# METHOD, APPARATUS, SYSTEM AND FIRMWARE FOR SECURE TRANSACTIONS

**Publication number:** WO9712344
**Publication date:** 1997-04-03
**Inventor:** CURRY STEPHEN M; LOOMIS DONALD W; FOX CHRISTOPHER W
**Applicant:** DALLAS SEMICONDUCTOR (US)
**Classification:**
**- international:** ***G06Q10/00; G06Q20/00; G06Q40/00; G06Q50/00; G07F7/08; G07F7/10; G09C1/00;*** **G06Q10/00; G06Q20/00; G06Q40/00; G06Q50/00; G07F7/08; G07F7/10; G09C1/00;** (IPC1-7): G07F7/10; G06F17/60; G07F19/00
**- European:** G07F7/08C2; G07F7/08C2B; G07F7/10D4E2; G07F7/10D4T; G07F7/10E
**Application number:** WO1996US15471 19960926
**Priority number(s):** US19950004510P 19950929; US19960594983 19960131

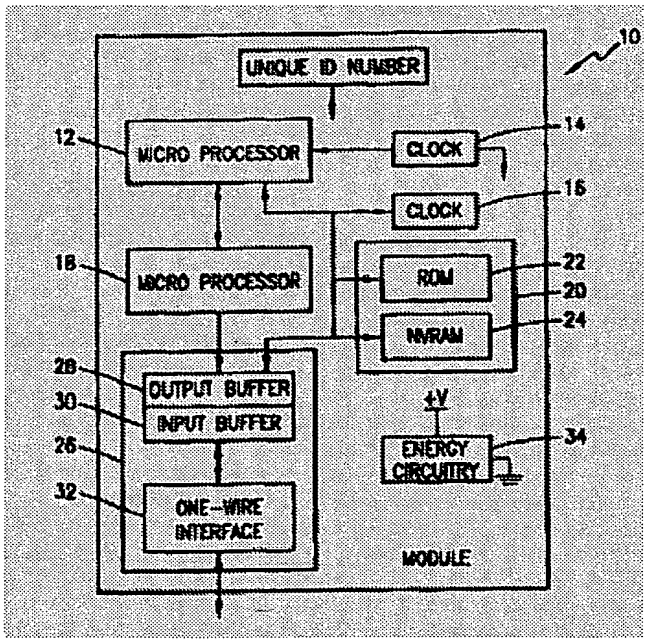
**Also published as:**
WO9712344 (A3
EP0862769 (A3
EP0862769 (A2
US6105013 (A1
US5748740 (A1
more >>

**Cited documents:**
EP0458306
EP0186981
EP0194839
DE4406602
EP0294248
more >>

**Report a data error he**

Abstract of **WO9712344**

The present invention relates to an electronic module used for secure transactions. More specifically, the electronic module is capable of passing information back and forth between a service provider's equipment via a secure, encrypted technique so that money and other valuable data can be securely passed electronically. The module is capable of being programmed, keeping track of real time, recording transactions for later review, and creating encryption key pairs.



(19)日本国特許庁（ＪＰ）　(12) **公表特許公報（Ａ）**　(11)特許出願公表番号

特表平11－513509

(43)公表日　平成11年(1999)11月16日

| (51)Int.Cl.[6] | | 識別記号 | FI | | |
|---|---|---|---|---|---|
| Ｇ０６Ｆ | 19/00 | | Ｇ０６Ｆ | 15/30 | ３６０ |
| | 17/60 | | Ｇ０７Ｆ | 7/10 | |
| Ｇ０７Ｆ | 7/10 | | Ｇ０９Ｃ | 1/00 | ６１０Ｃ |
| | 19/00 | | | | ６２０Ｂ |
| // Ｇ０９Ｃ | 1/00 | ６１０ | | | ６６０Ｂ |

審査請求　未請求　予備審査請求　有　（全 73 頁）　最終頁に続く

(21)出願番号　特願平9－513652
(86) (22)出願日　平成８年(1996)９月26日
(85)翻訳文提出日　平成10年(1998)３月30日
(86)国際出願番号　ＰＣＴ／ＵＳ９６／１５４７１
(87)国際公開番号　ＷＯ９７／１２３４４
(87)国際公開日　平成９年(1997)４月３日
(31)優先権主張番号　６０／００４，５１０
(32)優先日　1995年９月29日
(33)優先権主張国　米国（ＵＳ）
(31)優先権主張番号　０８／５９４，９８３
(32)優先日　1996年１月31日
(33)優先権主張国　米国（ＵＳ）

(71)出願人　ダラス　セミコンダクター　コーポレイション
アメリカ合衆国75244－3292　テキサス州，ダラス，サウス　ベルトウッド　パークウェイ　4401
(72)発明者　カーリイ，スチーブン，エム．
アメリカ合衆国75248　テキサス州　ダラス，クリアヘイブン　サークル　6646
(72)発明者　ルーミス，ドナルド，ダブリュ．
アメリカ合衆国75019　テキサス州　コッペル，ダコタ　レーン　316
(74)代理人　弁理士　浅村　皓　（外３名）

最終頁に続く

(54)【発明の名称】　安全取引のための方法、装置、システムおよびファームウェア

(57)【要約】
本発明は、安全取引に使用される電子モジュールに関する。より詳細には、この電子モジュールはマネーおよびその他の有価データを電子的に安全に送ることができるよう、安全な暗号化された技術によりサービスプロバイダの機器の間で情報を送ったり受信したりできる。このモジュールは後に検討できるように、リアルタイムのトラックを維持し、取引を記録し、暗号化鍵のペアを作成するようにプログラムできる。

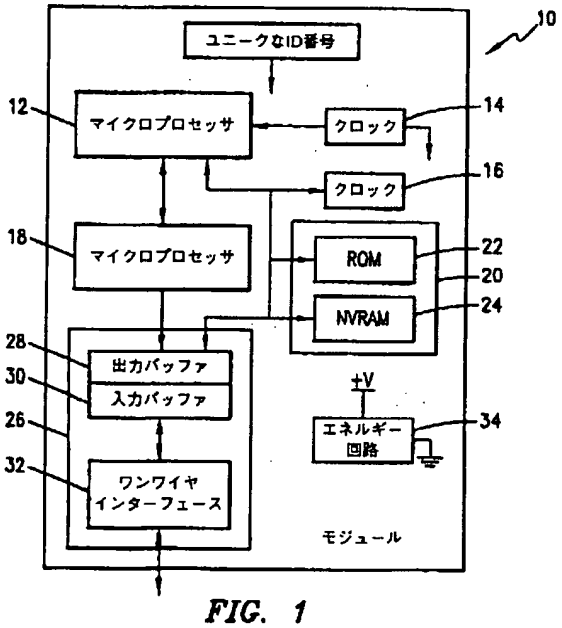



*FIG. 1*

【特許請求の範囲】

１．データ処理回路と通信するための入出力回路と、

前記入出力回路に電気的に接続された数値コプロセッサ回路と、

前記入出力回路に電気的に接続されたマイクロプロセッサ回路と、

前記マイクロプロセッサ回路に電気的に接続されたメモリ回路とを備え、前記データ処理回路との間で安全な暗号化されたデータの転送を行うようプログラム可能な、安全な取引を行うのに使用される電子モジュール。

２．前記データ処理回路が別の電子モジュールである、請求項１記載の電子モジュール。

３．前記入出力回路に接続されたワンワイヤインターフェースを更に備えた、請求項１記載の電子モジュール。

４．前記メモリ回路が前記電子モジュールと前記データ処理回路の間で暗号化されたデータを転送する間使用するための秘密暗号化／暗号解読鍵を記憶するようになっている、請求項１記載の電子モジュール。

５．前記暗号化された取引に時間スタンプが押される、請求項１記載の電子モジュール。

６．第１モジュールおよびサービスプロバイダの機器とを備え、

前記第１モジュールが、

入出力回路と、

乱数を発生するための乱数発生手段と、

乱数を発生し、この乱数を前記入出力回路に与えることを前記乱数発生手段に請求するための第１取引グループとを含み、

前記サービスプロバイダの機器が前記第１モジュールの前記入出力回路からの前記乱数を読み出すための手段と、

前記乱数と第１データとを組み合わせ、前記乱数と前記第１データとの組み合わせを秘密化鍵によって暗号化し、第１の証明書を作成するための手段とを備え、前記第１モジュールの前記入出力回路が前記第１の証明書を受信するようになっている、安全取引の通信を行うためのシステム。

７．前記サービスプロバイダの機器が第１のモジュールを含む、請求項６記載のシステム。

８．前記第１モジュールが、前記第１モジュールを識別するための識別子を更に備え、前記第１取引グループが前記識別子を前記入出力回路に与える請求項６記載のシステム。

９．前記読み出すための手段が前記第１モジュールの前記入出力回路からの前記識別子を更に読み出すようになっている、請求項８記載のシステム。

１０．前記第１モジュールが第２の取引グループを更に含む、請求項６記載のシステム。

１１．前記モジュールが完了した取引に時間のスタンプを押すための手段を更に含む、請求項６記載のシステム。

１２．モジュールとサービスプロバイダの機器との間で暗号化された情報を伝送するための方法において、

ａ）前記モジュールに第１の乱数を発生する工程と、

ｂ）前記乱数を前記サービスプロバイダの機器に送る工程と、

ｃ）少なくとも前記乱数を前記サービスプロバイダの機器内の秘密鍵により暗号化し、よって証明書を作成する工程と、

ｄ）少なくとも前記証明書を前記モジュールに送る工程と、

ｅ）前記証明書を前記モジュール内の公開鍵で暗号解読する工程と、

ｆ）前記第１乱数と、工程ｅ）の解読された第１の証明書内で発見された数字とを比較し、２つの数字が一致しているかどうかを判断する工程とを備えた、暗号化された情報を伝送するための方法。

１３．工程ｂ）がモジュールの識別子を前記サービスプロバイダの機器に送る工程を更に含む、請求項１２記載の方法。

１４．前記サービスプロバイダの機器が別のモジュールである、請求項１２記載の方法。

１５．前記方法が単線のバスを含む、請求項１２記載の方法。

１６．前記単線バスが実質的にワンワイヤバスである、請求項１５記載の方法

１７．モジュールとサービスプロバイダの機器との間で暗号化された情報を伝送するための方法において、

ａ）前記サービスプロバイダの機器内で第１の乱数を発生する工程と、

ｂ）前記乱数を前記モジュールに送る工程と、

ｃ）少なくとも前記乱数を前記モジュールの秘密鍵により暗号化し、よって第１の証明書を作成する工程と、

ｄ）少なくとも前記第１の証明書を前記サービスプロバイダの機器に送る工程と、

ｅ）前記第１の証明書を前記サービスプロバイダの機器内の公開鍵で暗号解読する工程と、

ｆ）前記第１乱数と、工程ｆ）の解読された第１の証明書内で発見された数字とを比較し、２つの数字が一致しているかどうかを判断する工程とを備えた、暗号化された情報を伝送するための方法。

１８．前記サービスプロバイダの機器が別のモジュールである、請求項１７記載の方法。

１９．前記方法が単線のバスを含む、請求項１７記載の方法。

２０．前記単線バスが実質的にワンワイヤバスである、請求項１７記載の方法。

２１．第１の暗号化されたデータおよび第２の暗号化されたデータを受信する工程と、

前記第１の暗号化されたデータを前記モジュール内に記憶された秘密鍵で解読し、第１の解読鍵を作成する工程と、

前記第１の解読鍵を電子システムに提供する工程と、

前記第２の暗号化されたデータを前記電子システムを介し、前記第１の解読鍵で解読し、よって有効な情報を作成する工程とを備えた、モジュールを使って暗号化されたデータを解読する方法。

２２．前記暗号化されたデータが電子メールメッセージである、請求項２１記載の方法。

【発明の詳細な説明】

安全取引のための方法、装置、システムおよびファームウェア

関連出願

本願は、１９９５年９月２９日に出願された米国仮特許出願第60/004,510号の利益を請求するものである。

出願人を本願出願人と同一とする下記の出願は関連する要旨を含み、よって参照によって本願の開示としてここに組み込む。

「値の単位を転送するための方法、装置およびシステム」を発明の名称とし、１９９６年１月３１日に出願された米国特許出願第08/595,014号。

「安全モジュールと別のモジュールとの間で有価情報を転送する方法」を発明の名称とし、１９９６年１月３１日に出願された米国特許出願第08/594,975号。

発明の背景

発明の技術分野

本願は、安全取引のために使用される方法、装置およびファームウェアに関する。特に電子モジュールに基づくシステムでは、少なくとも安全なデータ転送、デジタル署名を行い、またはマネー取引をオーソライズするよう、モジュールを構成できる。

関連技術の説明

現在ではクレジットカードに連動する磁気ストリップを有するクレジットカードは、市場における好ましいマネー取引媒体となっている。カードのユーザーはカードを自動キャッシュマシン、地域の店または銀行へ持って行き、マネー取引を行うことができる。多くの場合、マネー交換を行うのに電話のインターフェースを介してカードが使用される。カードおよびカードのユーザーの識別を助けるのに磁気ストリップカードが使用される。カードによる転送の安全性のレベルは比較的低い。それにも拘わらず、このカードにより、カードの所有者は製品を購入し、負債の支払いをし、異なる銀行口座間でマネーの交換を行うことができる。

これまでに磁気ストリップカードに関して改良がなされてきた。磁気ストリッ

プの代わりにマイクロ回路を備えたカードも作られており、一般にマイクロストリップのようにマイクロ回路を使用してカード読み取り機が取引を行うことができるようになっている。

発明の概要

本発明は好ましくは携帯モジュールとサービスプロバイダの機器との間で暗号化された情報をやり取りするための装置、システムおよび方法に関する。この発明は乱数、例えばＳＡＬＴを発生し、例えばマネー交換のためのリクエストと共にこの乱数をサービスプロバイダの機器へ送ることができる、ユニークな識別を有するモジュールを含む。次にサービスプロバイダの機器は他の情報と共に（取引のタイプに応じて）秘密鍵または公開鍵を使って乱数を符号化し、この符号化した情報をサインされた証明書としてモジュールへ送り戻すことができる。モジュールはサインされた証明書を受けると、（取引のタイプに応じ）公開鍵または秘密鍵を用いて証明書を解読し、解読した数字と元の乱数とを比較する。更に、両者の数字が同じであれば、リクエストされた取引を安全と見なし、取引を進める。これらモジュールは時間のスタンプを付けることができ、後に検討できるよう、取引に関する情報をメモリに記憶できる。

図面の簡単な説明

添付図面を参照し、次の詳細な説明を読めば、本発明の方法および装置についてより完全に理解できよう。

図１は、モジュールの一実施例のブロック図である。

図２は、取引グループを作成するためのプロセス例である。

図３は、Ｅメールメッセージを受けるための技術例である。

図４は、公証機能のためのモジュールを作成するための技術例である。

図５は、公証手段としてモジュールを使用するための技術例である。

図６は、マネー取引を行うためのモジュールを作成するための技術例である。

図７は、モジュールを使用してマネー取引を行うための技術例である。

図８は、モジュールを使用してマネー取引を行うための技術例である。

図９は、モジュールを使用してマネー取引を行うための技術例である。

図１０は、ネットワークを通してデータを送るための技術例である。

図１１は、モジュール内のソフトウェアおよびファームウェアの組織例である。

図１２は、モジュールにおけるソフトウェアおよびファームウェアの構成例である。

現在の時点で好ましい実施例の詳細な説明

図１は、本発明の一実施例を含むモジュール１０の一例のブロック図を示す。このモジュール回路は１つの集積回路とすることができる。このモジュール１０は組み合わせた多数の集積素子回路またはディスクリート素子回路で構成できると理解できよう。このモジュール１０はマイクロプロセッサ１２と、リアルタイムクロック１４と、制御回路１６と、数値コプロセッサ(math coprocessor)１８と、メモリ回路２０と、入出力回路２６と、エネルギー回路とを含む。

モジュール１０は限定するわけではないが、トークン、カード、リング、コンピュータ、財布、鍵ホルダー(Key fob)、バッジ、宝石、スタンプまたは対象のユーザーが持ったり、および／またはユーザーと連設できる、実質的に任意の物体を含む種々の物体内に組み込むことができるよう充分小さくできる。

マイクロプロセッサ１２は８ビットマイクロプロセッサであることが好ましいが、１６、３２、６４または任意の作動可能な数のビットのマイクロプロセッサとすることができる。クロック１４はモジュール回路に対するタイミングを定める。連続的に作動するリアルタイムクロックとなるような別個のクロック回路１４を設けてもよい。

数値コプロセッサ回路１８は極めて大きい数を処理するように設計されており、そのために使用される。特にこのコプロセッサはＲＳＡの暗号化および暗号解読の複雑な数学的処理を行う。

メモリ回路２０はリードオンリーメモリと、不揮発性のランダムアクセスメモリの双方を含むことができる。更に当業者であれば、等価的なデバイスを作るのに揮発性メモリ、ＥＰＲＯＭ、ＳＲＡＭおよび種々の他のタイプのメモリ回路を使用できることが理解できよう。

制御回路１６は回路全体に対するタイミング、ラッチングおよび種々の必要な制御機能を行う。

入出力回路２６はモジュール１０との双方向の通信を可能とし、この入出力回路２６は好ましくは少なくとも１つの出力バッファ２８と、入力バッファとを含む。ワンワイヤバスを介した通信を行うために、この入出力回路２６と共にワンワイヤインターフェース回路３２を設けることができる。

メモリ回路２０を維持し、および／またはモジュール１０内の他の回路への給電を助けるのに、エネルギー回路３４が必要である。このエネルギー回路３４はバッテリーと、コンデンサと、Ｒ／Ｃ回路と、光電池と、または他の等価的なエネルギーを発生する回路または手段から構成できる。

次に、安全取引モジュールおよびモジュール１０を使った一連のサンプルアプリケーションの好ましい実施例のファームウェアアーキテクチャについて説明する。これら例は、モジュール１０の好ましい特徴の組を説明し、かつモジュールが提供するサービスを説明しようとするものである。これらアプリケーションは本発明の機能を限定するものでなく、むしろその機能のサンプルを明らかにするものである。

Ｉ．好ましい実施例およびそのファームウェア設計の概観

モジュール１０は、好ましくは汎用の８０５１コンパチマイクロコントローラ１２または妥当な程度に同様な製品と、連続作動するリアルタイムのクロック１４と、大きな整数のための高速モジュラー累乗アクセレレータ（数値コプロセッサ）１８と、データを送受信するためのワンワイヤインターフェース３２を備えた入出力バッファ２８、３０と、あらかじめプログラムされたファームウェアを備えた３２キロバイトのＲＯＭメモリ２２と、重要なデータを記憶するための８キロバイトのＮＶＲＡＭ（不揮発性ＲＡＭ）２４と、入力回路２６内のデータを解釈し、これに作用するようマイクロコントローラ１２をパワーアップできる制御回路１６とを含む。モジュール１０はその作動電力をワンワイヤラインから引き出す。マイクロコントローラ１２と、クロック１４と、メモリ２０と、バッファ２８、３０と、ワンワイヤバスフロントエンド３２と、モジュラー累乗アクセレレータ１８と、制御回路１６は、好ましくは１つのシリコンチップ上に集積化され、データを破壊することなくＮＶＲＡＭ２４内のデータをさぐることを実質

的に不可能にさせるパッケージ技術を用い、ステンレススチール製のマイクロカン(microcan)内にパッケージされる。当初は下記に述べるようなアプリケーションをサポートするのに使用するためにＮＶＲＡＭ２４のほとんどを利用できる。当業者であればモジュールの構造の相当する変形例が多数あることが理解できよう。例えば揮発性メモリを使用したり、ワンワイヤバスインターフェース以外のインターフェースも使用できる。このシリコンチップはクレジットカード、リング等にパッケージできる。

モジュール１０は、好ましくは、最初は有効な機能を奏することができるよう、データをモジュール１０にロードするサービスプロバイダによって使用され、次にエンドユーザーの利益のためにサービスプロバイダに代って操作を行うためのコマンドをモジュール１０に送るエンドユーザーによって使用されるようになっている。このような理由から、モジュール１０は意図するアプリケーションのためにモジュールを設定する際に、サービスプロバイダをサポートするための機能を提供する。更にモジュールはサービスプロバイダが提供するサービスをエンドユーザーが受けることができるようにする機能も提供する。

各サービスプロバイダは取引グループ４０を作成することにより、そのサービスをサポートするためのＮＶＲＡＭメモリのブロックを保留できる（図１１～１２参照）。取引グループ４０とは単にサービスプロバイダによって定義される一組のオブジェクト４２である。これらオブジェクト４２としては、データオブジェクト（暗号化鍵、取引カウント数、金額、日／時間スタンプ）と、データオブジェクトを有効な方法でどのように組み合わせるかを指定する取引スクリプト４４の双方がある。各サービスプロバイダは他のどの取引グループ４０とも独立した自己の取引グループ４０を作成する。従って、多数のサービスプロバイダは同一のモジュール１０内で異なるサービスを提供できる。サポート可能な独立したサービスプロバイダの数は、各取引グループ４０内で定義されたオブジェクト４２の数および複雑さに応じて決まる。取引グループ４０内で定義できるオブジェクト４２の一部の例としては次のものがある。

ＲＳＡモジュール　　クロックオフセット

| | |
|---|---|
| ＲＳＡ指数 | ランダムＳＡＬＴ |
| 取引スクリプト | コンフィギュレーションデータ |
| 取引カウンタ | 入力データ |
| マネーレジスタ | 出力データ |
| デストラクタ | |

各取引グループ４０内ではモジュール１０は最初に取り消し不能な効果を有する所定のコマンドを取り込む。取引グループ４０内で一旦これら取り消し不能なコマンドのいずれかが実行されると、これらコマンドはモジュールの有効寿命が終了するまで、またはモジュールが使用する取引グループ４０がモジュール１０から削除されるまで、有効な状態のままになる。更にモジュールの寿命が終了するまで、またはモジュール１０の内容全体を消去するようマスター消去コマンドが発生するまで、取り消し不能な効果を有する所定のコマンドが残される。これらコマンドについては後に更に説明する。

これらコマンドはエンドユーザーが実行できる操作に対する必要な制御をサービスプロバイダに与えるのに基本的なコマンドである。取り消し不能なコマンドの一部の例としては次のものがある。

| | |
|---|---|
| 秘密化オブジェクト | ロックオブジェクト |
| ロック取引グループ | ロックマイクローインーＡーカン（商標） |

モジュールユーティリティの多くは、守秘能力を中心においているので、秘密化コマンドは極めて重要な取り消し不能なコマンドである。

モジュール１０が全体としてロックされると、残りのＮＶＲＡＭメモリ２４は今までの取引の監査トレイル（記録）を保持するための特定のバッファのために割り当てられる。取引の各々は取引グループの番号、指定グループ内の取引スクリプト４０の番号および日／時間スタンプによって識別される。

ファームウェアによって実行される基本概念は、サービスプロバイダがエンドユーザーに実行できるように望むオブジェクト間の動作だけを実行させるように、取引グループ４０内に取引スクリプトを記憶できるようにすることである。サービスプロバイダはサービスプロバイダに代ってモジュール１０が取引をサイン

できるようにするＲＳＡ鍵（暗号化鍵）を記憶し、秘密化し、よってその認証を保証することもできる。取引グループ４０内に１つ以上のオブジェクト４２を秘密化および／またはロック(lock ing)することにより、サービスプロバイダはサービスプロバイダに代ってモジュール１０が何をできるようにするかに関する制御

を維持する。エンドユーザーは新しい取引スクリプト４４を追加できないので、サービスプロバイダによってプログラムされた取引スクリプト４４により実行できるオブジェクト４２に対する動作に限定される。

ＩＩ．モジュールの利用モデル

この章は、最も簡単なことから最も複雑な範囲のモジュール１０の一連の実際のアプリケーションを示すものである。これらアプリケーションの各々は、どのような理由でモジュール１０が、このアプリケーションのための技術を可能にする中心となっているかを明らかにするために、充分詳細に説明する。

Ａ．安全Ｅメールの背景

この章では任意のロケーションで誰でも自分のＥメールを安全に受け取ることができるようにするのに、どのようにモジュール１０を使用できるかの一部を示す。

１．標準Ｅメール

標準Ｅメールシステムではユーザーのコンピュータはインターネットサービスのプロバイダに接続されており、ユーザーのコンピュータは新しいメールを求めてプロバイダのコンピュータを呼び出している際に、ユーザーのコンピュータはＥメールのパスワードを与える。Ｅメールは平文(plain text)のフォームでプロバイダのコンピュータに存在し、ここで働く者がＥメールを読む可能性がある。更に発信元から進む間、Ｅメールは多数のコンピュータを通過し、これらのロケーションで露出される。ユーザーがローカルエリアネットワークを通して自分のプロバイダから自分のメールを受け取った場合、同じネットワーク上の他の者がＥメールを捕捉し、読み取る可能性がある。最後に、ユーザーがパスワードを入力しなくてもすむような多数のＥメールシステムを使用した場合、ユーザーのコ

ンピュータ側にいる誰でもユーザーのメールを検索し、読み取ることができる。この理由は、ユーザーのコンピュータがプロバイダのコンピュータを呼び出しているとき、ユーザーのコンピュータは自動的にパスワードを送るからである。

ユーザーのコンピュータ内のコンフィギュレーションファイルからパスワードをコピーし、これを使って別のコンピュータからユーザーの命令を読み取ることもしばしば可能である。このように平文のフォームでのＥメールが広く分散し、

パスワード保護が弱い結果として、標準的なＥメールは極めて安全性が低いものと考えられる。

この問題を解消するために、Ｐ．Ｇ．Ｐ．(Pretty Good Privacy)として知られる安全システムが考えられた。このＰ．Ｇ．Ｐ．を使用するためにユーザーは公開成分および秘密成分の双方を含む完全なＲＳＡ鍵セットを発生する。ユーザーは自分のＥメールメッセージすべての署名ブロック内に公開鍵を挿入し、Ｐ．Ｇ．Ｐ．公開鍵の公にアクセス可能なダイレクトリ内にこの公開鍵を掲示することにより、ユーザーの公開鍵を広く利用できるようにする。ユーザーは多分パスワードで保護されたフォームで自分のパソコン内に自らの秘密鍵を記憶する。だれかがこのユーザーに秘密のＥメールを送りたい時にランダムＩＤＥＡ暗号化鍵を発生し、このＩＤＥＡ暗号化アルゴリズムを用いてメッセージ全体を暗号化する。次に、意図する受け手により提供される公開鍵を使ってＩＤＥＡ鍵自体を暗号化する。次に、ＩＤＥＡによって暗号化されたメッセージおよびユーザーの公開鍵を使って暗号化されたＩＤＥＡ鍵の双方をユーザーにＥメール送信する。この送信を見た者は、送ろうとする相手の受け手を除き、誰もこれを読むことはできない。その理由は、このメッセージはＩＤＥＡにより暗号化され、ＩＤＥＡ鍵は意図する受け手の公開鍵によって暗号化されているからである。受け手のコンピュータは対応する秘密鍵を記憶しているので、ＩＤＥＡ鍵を解読し、解読されたＩＤＥＡ鍵を使ってメッセージを解読できる。これにより遠隔からユーザーのメールを読み出そうと試みる者からの安全性が得られるが、ユーザーのコンピュータ自体が秘密鍵を記憶しているので、ユーザーのコンピュータが他のコンピュータにアクセス可能な場合には、これはあまり有効ではない。秘密鍵がパスワー

ドで保護されていても、ユーザーのパスワードを推定したり、またはパスワードの入力時にユーザーを盗み聞きすることは容易であることが多いので、ユーザーのコンピュータは安全性を低くしている。更にユーザーは自分のコンピュータでしか安全なＥメールを受信できない。その理由は、ユーザーの秘密鍵を記憶しているのは自分のコンピュータであり、ユーザーの秘密鍵は他の場所では利用できないからである。従って、Ｐ．Ｇ．Ｐ．の弱点は秘密鍵が存在するユーザーのコンピュータに強力に組み合わされていることである。

２．モジュールで保護されたＥメール

Ｅメールを保護するのにモジュール１０が使用されている場合、ユーザーはＥメールが他人によって読まれたり、または自分のパソコンがＥメールの安全性を危うくするような、弱いリンクとなるような恐れなく、ユーザーが行く場所でユーザーに自分のＥメールを送ることができる。このモジュールで保護されたＥメールシステムはＰ．Ｇ．Ｐ．システムと類似するが、パソコン内ではなく、モジュール１０の取引グループ内の秘密化オブジェクト内にＩＤＥＡ鍵を解読するのに使用される秘密鍵が記憶されている点が異なっている。このモジュールで保護されたＥメールシステムは次のように作動する。

ａ．図２、１１および１２を参照すると、ユーザーは取引グループ４０を作成し（Ｓ１）、ＲＳＡ鍵セットを発生し（Ｓ２）、これを取引グループ４０の３つのオブジェクト４２（１つはＲＳＡモジュールオブジェクトＮであり、２つはＲＳＡ指数(exponent)オブジェクトＥ，Ｄである）にロードする。ユーザーは次に解読指数Ｄを秘密化する（Ｓ３）。最後に、取引スクリプト４４を作成し（Ｓ４）、入力データオブジェクト内のデータを取り出し、これをモジュラスＮおよび秘密指数Ｄで暗号化し、この結果を出力データオブジェクト内に入れる。ユーザーは別の取引スクリプト４４が追加されないようにグループをロックする（Ｓ５）。ユーザーはＤの値を「忘れ」、公開デイレクトリおよび自分のＥメールメッセージの署名ブロック内にＥおよびＮの値を公開する。ユーザーはＤを忘れており、Ｄの指数オブジェクトは秘密にされているので、だれかがＤの値を見つけることはない。

ｂ．図３を参照すると、ユーザーに安全なＥメールを送るのにＰ．Ｇ．Ｐ．システムが使用される。ユーザーが安全なＥメールを受けると（Ａ１）、ユーザーは暗号化されたＩＤＥＡ鍵を取引グループ４０の入力データオブジェクトへ送り（Ａ２）、次に取引スクリプト４４をコールし、この鍵を解読し（Ａ３）、解読された結果を出力データオブジェクトへ入れる（Ａ４）。次にユーザーは、出力データオブジェクトから解読されたＩＤＥＡ鍵を読み出し、これを使って自分のメールを解読する（Ａ５）。ここで、モジュール１０を物理的に所有することなく、ユーザーを含むだれかが新しいメールを読むことは、この時点では不可

能であることに留意されたい。従って、Ｅメールを読むコンピュータにはモジュール１０が物理的に存在していなければならないので、ユーザーに無断でユーザーのメールを読み出す方法はない。ユーザーは自分が行く場所に自分のモジュール１０を携帯し、これを使ってどんな場所でも、送られて来たメールを読むことができる。このように、ユーザーの家のコンピュータが安全システムにおける弱点とはならない。

上記のような安全なＥメールは、１つのＲＳＡ鍵と１つの取引スクリプト４４しか必要としない最も簡単な、可能なモジュールアプリケーションである。モジュール１０内に公開鍵Ｅを記憶することも必要がなく、公開鍵は公にアクセス可能と考えられるので、このようにすることはうまいやり方である。指数オブジェクト内に公開鍵Ｅを記憶し、そのオブジェクトすなわちモジュラスオブジェクトＮを秘密化しないことにより、ユーザーはモジュール１０から常時公開鍵を読み出すことができるように保証できる。モジュール１０は暗号化を実行する必要はないので、公開鍵Ｅに関連する取引スクリプト４４はない。

Ｂ．デジタル公証サービス

この章は、モジュール１０を使った好ましい公証サービスを説明するものである。

１．標準公証サービスの背景

従来の公証サービスプロバイダはエンドユーザーからの文章を受信し、これを

検査し、所定の日に公証人にその文書が提示されたことを証明する偽造不能なマークを文書に付ける。かかる公証サービスの１つの利用法として、必要な場合に裁判所で発明の優先権を後に主張できるよう、新しい発明の開示を記録することがある。この場合、公証人によって提供される最も重要なサービスは、所定の日に発明者の所有物としてその開示物が存在していたことを証明することである。（優先権を決める従来の方法は、重要な発明の開示物に発明者と証人がサインし、日付を記載した研究ノートを用いることである。）

２．モジュールを使用した電子公証サービス

会社（以下サービスプロバイダと称す）は、顧客（以下エンドユーザーと称す）に対する公証サービス（厳密には優先権証明サービス）を行うビジネスに

参入すると決定する。このサービスプロバイダはモジュール１０をそのエージェントとして使用することにより、このサービスを行うことと、サービスプロバイダに代って文章を認証する（日付を記載し、サインする）権限をエージェントに与える。このシステムの好ましい作動は次のとおりである。

ａ．図４、１１および１２を参照すると、サービスプロバイダはモジュール１０の「登録されたロット」における電子公証機能を行うための取引グループ４０を作成する（Ｂ１）。

ｂ．サービスプロバイダは安全な計算施設を使ってＲＳＡ鍵セットを発生し、３つのオブジェクト４２のセット、すなわちモジュラスオブジェクトと２つの指数オブジェクトとして各モジュール１０へＲＳＡ鍵セットをプログラムする（Ｂ２）。鍵セットの公開部分はできるだけ広く知られるようにされており、秘密部分はサービスプロバイダによって完全に忘れられている。秘密指数オブジェクトはモジュール１０から読み戻されることがないように秘密にされる。

ｃ．サービスプロバイダは各モジュール１０からリアルタイムクロック１４を読み出し、リアルタイムクロック１４の読み出しと適当な基準時間（例えば１９７０年1月1日１２：００ａｍ）との間の差を含むクロックオフセットオブジェクトを作成する。リアルタイムクロックにこのクロックオフセットオブジェクトの値を加えることにより、任意のモジュール１０から真の時間を得るこ

とができる（Ｂ３）。

ｄ．サービスプロバイダは０に初期化された取引シーケンスカウンタオブジェクトを作成する（Ｂ４）。

ｅ．サービスプロバイダは取引カウンタの値が続き、次にユニークなレーザー登録番号が続く真の時間（クロック１４の真のクロックと、クロックオフセットオブジェクトの値の合計）に入力データオブジェクトの内容を添付する、取引スクリプト４４を作成する。次にこの取引スクリプト４４はこのデータのすべてを秘密鍵で暗号化し、出力データオブジェクト内に入れることを指定する。この動作を実行させる命令は、取引スクリプトオブジェクトとして取引グループ４０内に記憶される（Ｂ５）。

ｆ．サービスプロバイダは直接読み取り可能、または書き込み可能とならないように望む他のオブジェクト４２を秘密化する（Ｂ６）。

ｇ．サービスプロバイダは取引グループ４０をロックし、別の取引スクリプト４４が追加されるのを防止する（Ｂ７）。

ｈ．図５を参照すると、次にサービスプロバイダはモジュールを支払い側の顧客（エンドユーザー）に分配し公証サービスに使用する。エンドユーザーが文書を証明させたい時はいつも、エンドユーザーは（安全ハッシュ規格(Secure Hash Standard)、ＦＩＰＳ公開番号１８０に指定された）安全ハッシュアルゴリズムを実行し、文書全体を２０バイトメッセージのダイジェストにまとめる。エンドユーザーは２０バイトのメッセージダイジェストを入力データオブジェクトへ送信し（Ｃ１）、メッセージダイジェストを、真の時間と、取引カウンタと、ユニークなレーザーシリアル番号と、組み合わせること及びその結果得られたパケットに秘密キーでサインすることを、取引スクリプト４４に要求する（Ｃ２）。

ｉ．エンドユーザーは証明書を公開鍵で解読し、メッセージダイジェスト、真の時間スタンプ等をチェックし、これらが正しいかどうかを確認するように証明書をチェックする（Ｃ３）。次に、エンドユーザーはデジタルフォームで文書のオリジナルコピーと共にこのデジタル証明書を記憶する。サービスプロ

バイダはそのモジュールによって作成された証明書の認証を確認する。

ｊ．サービスプロバイダの指定した時間の後、ユーザーは自分のモジュール１０を戻し、料金を払い、新しい秘密鍵を含む新しいモジュールを得る。古いモジュールは取引グループ全体を消去し、これらを再プログラミングすることによりリサイクルできる。サービスプロバイダはこれまで使用した公開鍵のすべてのアーカイブを維持しているので、古い証明書の認証を必要に応じて証明できる。

Ｃ．デジタルキャッシュディスペンサ

この使用モデル例は、商品またはサービスに対する支払いを行うことができるキャッシュリザーバとしてのモジュール１０に焦点を合わせたものである（説明を簡単にするためにモジュール１０にキャッシュを補充することについては、後まで説明を延ばす）。この場合、サービスプロバイダは銀行または金融機関であり、エンドユーザーは購入のためにモジュール１０を使用したいと考える銀行の顧客であり、業者は購入された商品またはサービスの提供者である。これら取引におけるサービスプロバイダ、業者およびエンドユーザーの役割については後に詳細に説明する。

モジュール１０内で実行されるデジタルキャッシュ財布の基本概念は、モジュール１０がまず所定のキャッシュ値を含むロックされたマネーオブジェクトを含み、モジュール１０がマネーオブジェクトの値からリクエストされた金額を減算した事実を証明する、基本的にサインされた文書である証明証を要求に応じて発生できる概念である。これらサインされた文書は証明書の値に対応する金額だけ内部マネーオブジェクトの値が減少された事実を証明するので、現金に等価的な文書となっている。業者はこれら証明書をサービスプロバイダに戻すことにより、現金でこれら証明書を買い戻すことができる。

現金を表示するデジタル証明書を処理する際に、「再生」(replay)または複製が可能であることが基本的な問題となる。デジタルデータは容易にコピーし、再送信できるので、製造時に使用される特殊な技術によって複製が困難な通常の貨幣または紙幣と異なっている。このような理由から、支払いの受け手は受け取る

デジタル証明書がこれまで発行された証明書の複製でないことを保証するために、特殊な工程をとらなければならない。この問題は、受取人にランダム「ＳＡＬＴ」、すなわちチャレンジ番号を発生させ、支払い人にこれを提供させることによって解決できる。

ＳＡＬＴは再生を防止する１つの方法である。チャレンジ／レスポンスモードで乱数を送り、これを使用する。他方の当事者はこれらのレスポンスの一部として乱数を戻すようにチャレンジされる。

支払い人は金額および受取人のＳＡＬＴの双方を含むサインされた証明書を作成する。受取人がこの証明書を受けると、受取人は公開鍵でこれを解読し、金額をチェックし、支払い人が提供したものとＳＡＬＴが同じであることを確認する。受取人に対し証明書を秘密にすることにより、支払い人は証明書が複製又は再生でなく、正当なものであることを受取人に対して証明する。この方法は、モジュール１０が支払い人か受取人かによらず、使用できる。

解決しなければならない別の問題は、支払いの拒絶ができないことである。このことは、取引への当事者のいずれも、自分が実際に取引に参加していなかったと主張できないことを意味する。取引の記録（マネー証明書）には、取引への各当事者が意図的な参加者であったことを証明する要素が含まれていなければならない。

１．従来の現金取引の背景

従来の現金取引では、エンドユーザーは、まず銀行からの米国連邦準備ノート（紙幣）を受け、銀行はユーザーの口座の預金残高から等価的な金額を差し引く。エンドユーザーは下記の事項を含む公開鍵を使って連邦準備ノート（紙幣）の正当性を証明できる。

ａ．磁気によって吸引できる磁気インク

ｂ．紙幣にすき込まれた赤と青の繊維

ｃ．グラビア印刷された肖像を囲む微小な印刷

ｄ．米国および紙幣の表示と共に印刷された、埋め込まれたストライプ

このシステムに対する「秘密鍵」は、マネーを印刷するための未処理材料をどのように得て、どのようにマネーを実際に印刷するかの詳細である。このような情報は、政府によって保持されており、明らかにされていない。

これらの紙幣はエンドユーザーによって業者まで運ばれ、商品またはサービスと交換される。業者も紙幣の公開鍵を使って紙幣が正当なものであることを立証する。

最後に業者は紙幣を銀行へ持って行き、ここで支払機によって再び公開鍵が検査される。紙幣が正当なものであれば、紙幣の額面の値だけ業者の銀行口座の預金残高の金額が増す。

このような取引の最終結果として、エンドユーザーの銀行の預金残高が減少され、同じ金額だけ業者の銀行の預金残高が増やされ、商品またはサービスが業者からエンドユーザーへ転送され、銀行準備ノートを他の取引にすぐに再使用できる。

２．モジュールを使ったマネー取引の例

モジュール１０を使ったマネー取引およびデジタル証明書が連邦準備ノートと異なりデジタルデータを容易にコピーし、複製できるので、多少複雑となっている。それにも拘わらずＳＡＬＴおよび取引シーケンス番号を使用することにより、デジタル証明書の正当性を保証できる。（次の説明では、取引への各当事者は秘密に維持できる秘密鍵と自分のＲＳＡ鍵セットを有すると仮定する。）

ａ．図６を参照すると、サービスプロバイダ（銀行）はモジュール１０内に記憶されたマネーの値を示すマネーオブジェクトを含む取引グループ４０を作成することにより、モジュール１０を作成する。サービスプロバイダは取引カウントオブジェクト、モジュラスオブジェクトおよび指数オブジェクトも作成し、指数オブジェクトにプロバイダの秘密鍵を記憶する（Ｄ１）。サービスプロバイダは読み取りできないように鍵を秘密にする（Ｄ２）。次に、マネー取引を実行するように取引グループ４０内に取引スクリプト４４を記憶し、マネー取引を実行し、別のオブジェクトを作成できないように（Ｄ３）グループをブロックする（Ｄ４）。（この取引スクリプトが何を行うかの詳細については、下記で更

に説明する。）最後に、誰もが公開鍵を得ることができるようにサービスプロバイダが対応する公開鍵を公開する。

ｂ．エンドユーザーはサービスプロバイダからモジュール１０を受け、エンドユーザーの銀行口座ではモジュール１０に記憶された金額が貸方に記載される。パソコンまたはハンドヘルドコンピュータを使い、エンドユーザーは預金残高が正しいかどうかを確かめるためにモジュール１０に問い合わせできる。

ｃ．図７を参照する。エンドユーザーが業者からのある商品またはサービスを受けたい時（Ｅ１）、業者はモジュールのユニークな、レーザー食刻(lasered)登録番号を読み出し、これをランダムＳＡＬＴと共にパケットに入れる（Ｅ２、Ｅ３）。次に業者は自分の秘密鍵でこのパケットにサインをし（Ｅ４）、購入金額と共に、その結果暗号化されたパケットを取引グループ４０の入力データオブジェクトへ送信する（Ｅ５）。

ｄ．業者は次に、サービスプロバイダによってモジュール１０内にプログラムされた取引スクリプト４４を発動させる。この取引スクリプト４４はマネーオブジェクトから購入金額を差し引き（Ｅ６）、取引カウンタオブジェクトの値を入力データオブジェクトの内容に添付し（Ｅ７）、生じたパケットに秘密鍵でサインをし、その結果を出力データオブジェクトに入れる（Ｅ８）。

ｅ．業者は出力データオブジェクトから結果を読み出し、これをサービスプロバイダの公開鍵で解読する（Ｅ９）。次に購入金額が正しいこと、および残りのデータが工程ｃでサインしたパケットに同一であることを確認する（Ｅ１０）。

ｆ．モジュール１０で提供された証明書が正当かつオリジナルであること（複製でないこと）を確認した後に、業者は商品またはサービスを発送する（Ｅ１１）。その後、業者はデジタル証明書を銀行へ送る。

ｇ．銀行はサービスプロバイダの公開鍵を用いて証明書を解読し（Ｅ１２）、購入金額および取引カウントを抽出し、業者の公開鍵を使って残りのデータを解読し、モジュールのユニークな、レーザー食刻登録番号を明らかにする（Ｅ１４）。銀行は次にデータベース内のユニークなレーザー食刻登録番号を使

ってモジュール１０をルックアップし、この取引のための取引カウントが以前送られたものでないことを確認する。このテストに通ると、銀行はデータベースに取引カウント値を加え、次に購入金額だけ業者の銀行の預金残高を増す（Ｅ１５）。証明書のこの部分がモジュール１０および業者の双方によってサインされた事実により、取引が業者とモジュール１０の双方が任意に同意したものであることが確認される。

取引カウンタ値、ユニークなレーザー食刻登録番号、受取人により提供されるランダムＳＡＬＴおよびモジュールの秘密鍵、業者の秘密鍵またはその双方によって暗号化される購入金額のデータを組み合わせる異なる方法は多数あることに留意されたい。これら組み合わせの多くも、ユニークさ、本物であることおよび支払い拒否できないことを満足できるように保証できるものであり、ファームウェアの設計は取引スクリプト４４を作成する際のサービスプロバイダのフレキシビリティによってプロバイダの特定のニーズを満たすことができる。

Ｄ．デジタル現金補充

上記ＩＩ．Ｃ章ではデジタル現金財布について説明したが、現金の補充の問題については触れていない。サービスプロバイダは別のモジュラスオブジェクトおよびサービスプロバイダの公開鍵、ランダムＳＡＬＴオブジェクトおよび預金残高にマネーを加えるための取引スクリプト４４を含む指数オブジェクトを加えるだけで、ＩＩ．Ｃ章で説明したようにモジュール１０に現金補充能力を加えることができる。サービスプロバイダは直に、またはネットワークを通して遠隔からモジュール１０へマネーを加えることができる。このマネーを加えるプロセスは次のとおりである。

１．図８を参照すると、サービスプロバイダはモジュールのユニークなレーザー食刻登録番号（ＩＤ番号）を読み出し（Ｆ１、Ｆ２）、ランダムＳＡＬＴオブジェクトの値を戻すよう取引スクリプト４４に要求する。モジュール１０は先の値および乱数発生器から新しいランダムＳＡＬＴの値を計算し、これをサービスプロバイダに戻す（Ｆ３）。

２．サービスプロバイダは加えるべきマネーの金額およびモジュール１０

のユニークなレーザー食刻登録番号と共にモジュール１０によって戻されたランダムＳＡＬＴをパケットに入れ、この結果得られたパケットをサービスプロバイダの秘密鍵を用いて暗号化する（Ｆ４）。このような暗号化されたパケットは取引グループ４０の入力データオブジェクト内に書き戻される。

３．サービスプロバイダの公開鍵を用いて入力データオブジェクトの内容を解読する取引スクリプト４４を要求し、ユニークなレーザー食刻登録番号およびランダムＳＡＬＴの値をサービスプロバイダが最初に与えたものに対してチェックする。ＳＡＬＴが一致していれば、パケットがマネー金額を抽出し、これをモジュール内のマネーオブジェクトの値に加える（Ｆ５）。ユニークなレーザーを使った登録番号が含まれていることは厳密には必要でないが、サービスプロバイダがファンドをどのモジュールが受けているかを正確に知ることができるようにこの登録番号が含まれていることに留意されたい。

Ｅ．モジュール間のファンドの直接送金の説明

上記ＩＩ．Ｃ．２．ｇ章では業者がデジタル証明書を自分の銀行へ戻し、業者の口座に対する貸方に記入する際に問題が生じることを示した。業者側の銀行は払い戻しのためにサービスプロバイダに証明書を送り戻すか、または取引カウントオブジェクトの値がユニークであるかどうかを判断できるように、データベース内のサービスプロバイダの記録にアクセスしなければならない。このことは不便であり、これを行うためのインフラストラクチャを必要とする。また、業者側の銀行は使用済みの証明書の番号が再使用されないように、この証明書の番号をデータベースにロギングしなければならないので、このことは取引のいずれかが匿名（現金が既に使用された場合になるように）となることも防止する。これら問題は、モジュール間のファンド送金を活用することによりすべて除くことができる。更に、モジュール間のファンド送金を行うのに必要な工程は、ＩＩ．Ｃ．２章に記載されている工程よりもかなり簡単である。

次の説明では、業者もエンドユーザー（顧客）から受け取ったファンドを集めるのに使用するモジュールを有する。以下、エンドユーザーの所有するモジュールを「支払い人」と称し、業者の所有するモジュールを「受取人」と称す。ファ

ンド送金を行うための工程は次のとおりである。

１．図９、１１および１２を参照する。業者は自分のコンピュータを使用して受取人内の取引スクリプト４４に要求して、ランダムＳＡＬＴを提供させる。業者は取引グループ４０の出力オブジェクトからこのＳＡＬＴを読み出す。

２．業者はこのＳＡＬＴおよびエンドユーザーの購入金額を支払い人の入力データオブジェクトにコピーし（Ｇ１）、支払い人内の取引スクリプト４４に要求して、預金残高から購入金額を差し引き、パケット内の受取人のＳＡＬＴと購入金額とを組み合わせ、この結果得られたパッケージをサービスプロバイダの秘密鍵で暗号化し、これを出力データオブジェクトに戻させる（Ｇ２）。

３．次に業者はこのパケットを読み出し、これを受取人の入力データオブジェクトにコピーし、受取人内の取引スクリプト４４に要求してパケットをサービスプロバイダの公開鍵で解読し（Ｇ３）、ＳＡＬＴを、受取人が最初に発生したものと照合する。これらが一致していれば、受取人は購入金額を預金残高に加える（Ｇ４）。

これによりファンド送金が完了する。この取引は支払い人から受取人へ購入金額を有効に転送し、取引の工程はＩＩ．Ｃ．２に記載した３方向取引よりもかなり簡単であることに留意されたい。銀行がＳＡＬＴを業者のモジュールへ送り、業者が銀行へ送った預金残高の証明書を業者のモジュールが作成する同様な取引

により、業者は自分の銀行口座へ預金残高を転送できる。ファンドを集めるために業者がモジュールを使用することにより取引が簡略になり、ユニークさを確認するためのデータベースが不要となり、通常、キャッシュ取引で生じるようなエンドユーザーの匿名性が留保できる。

Ｆ．ネットワークを通したモジュールを使った取引例

上記ＩＩ．Ｃ．２、ＩＩ．ＤおよびＩＩ．Ｅ章に記載の取引は、ネットワークを通しても行うことができ、これにより業者とエンドユーザーとモジュールとを物理的に分離できる。しかしながらこれにより、モジュール１０に対する組み合わせのうちの１つが暗号化されず、よって偽造が行われ得るので、このようなネットワークを通した取引は潜在的な問題を生じ得る。この問題を回避するため、

双方の当事者は１つのＳＡＬＴを発生し、他方の当事者がサービスプロバイダの秘密鍵を使ってＳＡＬＴを暗号化できることを実証し、よって正当性を証明しなければならない。このような操作はモジュール間のファンドの送金（上記ＩＩ．Ｅ章）に関連するように、次のように説明される。この方法は、上記取引のうちの任意のものがネットワークを通して行うことができるように使用できる。これにより、明らかにインターネットを通した安全な電子的な取引が可能となる。

１．図１０、１１および１２を参照する。支払い人はランダムＳＡＬＴを発生し、ネットワークを通してこれを受取人に送信する（Ｈ１）。

２．受取人は支払い人のＳＡＬＴに購入金額を加え、その後に受取人によってランダムに発生されたＳＡＬＴが来る。次に受取人はこのパケットをサービスプロバイダの秘密鍵で暗号化し、これを支払い人に戻す（Ｈ２）。

３．支払い人はパケットをサービスプロバイダの公開鍵で解読し（Ｈ３）、支払い人のＳＡＬＴを抽出し、これを支払い人が工程１で与えたＳＡＬＴと比較する。これらが一致していれば、支払い人は預金残高から購入金額を差し引き、購入金額とサービスプロバイダの秘密鍵で暗号化した受取人のＳＡＬＴからなる証明書を発生し、これを受取人へ戻す（Ｈ５）。

４．受取人はパケットをサービスプロバイダの公開鍵で解読し（Ｈ６）、受取人のＳＡＬＴを抽出し、これと、受取人が工程２で与えたＳＡＬＴとを比較し、これらが一致していれば、受取人は自分の預金残高に購入金額を加える（Ｈ７）。

ＳＡＬＴの交換により各モジュールは他のモジュールと通信していること、よってリクエストされたファンド送金が正当なものであることを確認できる。工程３で述べたＳＡＬＴの比較により、支払い人はファンドを引き出す前に受取人が正当なモジュール１０であることを確認でき、工程４に記載の比較により受取人はファンドを入金する前に支払い人が正当なモジュール１０であることを確認できる。上記取引により、暗号化されたパケット内の必要な最小量の情報が得られ、よってファンドが１つのモジュール１０から他のモジュールに送金されていることを確認できる。付加情報を提供しそして取引をより完備するために、ユニー

クな、レーザー食刻登録番号のような他の情報を（匿名性を犠牲にして）含ませることもできる。

G．ソフトウェアの正当化および使用量の計量のための技術例

モジュール１０は包括的なソフトウェアシステムにおける特定のソフトウェアの特徴を可能にするタスクおよびこれら特徴の利用を計量化するために良好に適している。（この使用モデルはモジュール１０からマネーを引き出すための、これまで説明したモデルに匹敵するものである。）

１．準備

図１１および１２を参照する。サービスプロバイダは取引グループ４０を作成し、エンドユーザーがモジュール１０内のどのソフトウェアを使用するのかが許可されるかを詳細に定めるグループ内にコンフィギュレーションオブジェクトを記憶する。更に許可された使用クレジット（これは実際のドルの金額ではなく、時間単位とすることができる）を含むマネーオブジェクトも作成し、正当化のために使用するよう、秘密ＲＳＡ鍵ペアを記憶し、これを秘密にする。取引スクリプト４４を記憶し、ＳＡＬＴおよび金額を受け取り、エンドユーザーから引き出し、引き出した金額だけ預金残高を引き下げ、引いた金額、販売およびコンフィギュレーションオブジェクトの値を含むＲＳＡでサインされた証明書を出力する。

２．使用

モジュール１０内のソフトウェアを使用する間の周期的なインターバルで、パソコンのプログラムはランダムＳＡＬＴおよびモジュール１０の使用に対する料金の金額を発生し、この情報をモジュール１０へ送信する。モジュール１０は預金残高をデクリメントし、証明書を戻す。パソコンは証明書を解読し、ＳＡＬＴが同一であり、引いた金額が正しく、コンフィギュレーションオブジェクト内に記憶されている情報によりモジュール１０内のソフトウェアの使用が正当化されたことを確認する。これらの検査のいずれにも成功すると、モジュール１０は別の証明書がモジュール１０に求められるまで、所定の時間の間、または所定の操作回数だけ実行する。

この使用モデルには多数の変形例が可能になる。例えば取引スクリプト４４はパソコンで実行されるアプリケーションプログラムが実行時間を正確に測定することを保証できるよう、証明書内に真の時間をバインドすることも可能である。（これを行うには、測定時間の基準を与えるための初期化中にサービスプロバイダがクロックオフセットオブジェクトを発生しなければならない。）

H．取引タッチメモリ（商標）のシミュレーション

この使用モデルは、より簡単な取引タッチメモリ（商標）（ＤＳ１９６２年）（以下ＴＴＭとする）またはこれと均等または同様に作動できる同様なデバイスまたは置換物の作動をシミュレートするのにモジュール１０をどのように使用できるかを述べるものである。ＴＴＭの基本的な特徴は、メモリブロックの内容が変更された時はいつも、カウンタを自動的にインクリメントするように、メモリのブロックと連動したカウンタが設けられていることである。

１．準備

この簡単な特徴は、コンフィギュレーションオブジェクトと、取引カウンタオブジェクトと、取引スクリプトオブジェクト（このオブジェクトは入力オブジェクトの内容と取引カウンタオブジェクトの値を組み合わせ、これらをコンフィギュレーションオブジェクトに入れる）を作成し、プロセス内でカウンタを自動的にインクリメントすることにより、モジュール１０にプログラムできる。３つのすべてのオブジェクト４２はロックされるが、いずれも秘密にはされない。

２．使用

マネーを加えたり、引き出したりするのに、エンドユーザーはコンフィギュレーションオブジェクトおよび取引カウンタオブジェクトの値を直接読み出し、コンフィギュレーションオブジェクトを解読し、解読されたパッケージからの取引カウントとカウンタオブジェクトの値とを照合する。エンドユーザーは暗号化されたパケットからのユニークな、レーザー食刻登録番号とモジュール１０の登録番号とを照合する。これらが一致していれば預金残高を有効と見なす。預金残高から所定の金額を加えたり、差し引きし、取引カウントをインクリメントし、パケットを再び暗号化し、これを入力データオブジェクトに記憶する。次にデータ

および取引カウンタ値をコンフィギュレーションオブジェクトに移すのに取引スクリプト４４が要求され、プロセス内で自動的にカウンタ値をインクリメントする。（コンフィギュレーションオブジェクト内の値が変更されるたびに、取引スクリプト４４はカウンタオブジェクトの値がインクリメントされることを保証する。）

このような簡単な操作は、モジュール１０が暗号化自体を実行する必要がないので、比較的短時間で実行できる。しかしながらＴＴＭを用いた場合のように、エンドユーザーは次に暗号化および解読操作を実行するのに安全な計算施設を使用しなければならない。従って、このような使用はモジュールの暗号化機能に依存したものよりも低いレベルに保護される。

Ｉ．郵便計量サービスのための技術例

この使用モデルはモジュール１０を使って郵便証明書を送り出すアプリケーションを説明するものである。証明書を構成するデジタル情報は二次元のバーコードとして封筒上プリントされ、このバーコードはサービスプロバイダによって読み出され、認証できる（Ｕ．Ｓ．Ｐ．Ｓ．）。モジュール１０と組み合わされてレーザープリンタに接続された通常のパソコンで作動するコンピュータプログラムを使って郵便証明書をプリントできる。

１．準備

サービスプロバイダはマネーレジスタ、どのモジュールにも共通な秘密ＲＳＡ鍵（指数オブジェクトおよびモジュラスオブジェクト）と取引スクリプト４４を含むグループを作成する。スクリプト４４はＳＡＬＴおよび（エンドユーザーのコンピュータによって与えられる）引出し金額とモジュール１０のユニークな、レーザー食刻登録番号とを組み合わせ、このパケットを秘密鍵で暗号化し、預金残高から金額を差し引き、出力オブジェクト内に暗号化された証明書を入れ、ここで証明書をパソコンで読み出すことができる。

サービスプロバイダは所定のマネー金額で預金残高を初期化し、預金残高およびスクリプト４４をロックし、ＲＳＡ鍵オブジェクトを秘密にし、それ以上スクリプトを加えることができないようにグループをロックする。このように作成さ

れたモジュールをパソコンに基づく郵便計量プログラムと共に使用できるように、カウンタを通して販売できる。

２．使用

最初の封筒をプリントすべき際に、パソコンプログラムは日および一部のユニークなレーザーを使った登録番号の一方向ハッシュ（例えば安全ハッシュ規格、ＦＩＢＳ　ＰＵＢ　１８０）を計算することにより、最初のＳＡＬＴを作成する。この情報は、引き出すべき郵便量と共にモジュール１０へ送られる。この結果生じた証明書はハッシュ発生番号（最初のハッシュに対しては１つ）、ユニークなレーザー食刻登録番号、スタンプの平文表示、日およびエンドユーザーを識別したい場合の他の情報と共に、二次元バーコードとしてプリントされる。その後のＳＡＬＴは先のＳＡＬＴに対し一方向ハッシュを再び実行し、ハッシュ発生番号をインクリメントすることによって発生される。

サービスプロバイダが封筒を受けると、封筒のほとんどは額面の値が読み取られ、デジタルバーコードは読み取られない。しかしながらバーコードの統計的なサンプリングを読み取り、得られた情報を公開鍵で解読し、証明する。食い違いを調査し、現行法により偽造を追求する。サンプリングはユニークなレーザー食刻登録番号、日付およびハッシュ発生番号からＳＡＬＴを再作成し、取引が現在のものであるだけでなく、特定のモジュール１０にリンクしていることを立証できる。

同様な結果を生じさせる上記方法には、可能な変形例は多数ある。最もあり得る偽造は複製であり、これはユーザーがプリンタへ送るデジタル情報を捕捉し、郵便証明書を作成し、証明書と同じ複製コピーを多数作成する偽造である。このような偽造はハッシュ発生番号およびユニークな登録番号を読み出し、これらをデータベースでルックアップし、ユーザーが同じ証明書を複製しているかどうか

を確認するだけで、サービスプロバイダはこの複製を容易に見つけることができる。（このようなチェックはＲＳＡ解読が必要な完全な証明書の立証よりも頻繁に行うことができる。）

Ｊ．加入情報サービス

この使用モデルは利用可能な情報に対し支払いを行うことに同意したユーザーに対し、インターネットを通して暗号化されたフォームの利用可能な情報をサービスプロバイダが作成するアプリケーションについて述べたものである。このアプリケーションは上記Ａ章で述べた安全なＥメール使用モデルと全く同じように働くが、サービスプロバイダがユーザーにＥメールで送った暗号化された情報に対し、ユーザーに料金を請求する点が異なっている。この課金情報は、サービスプロバイダの公開鍵またはサービスプロバイダのモジュール１０のユニークなレーザーを使ったシリアル番号に基づき、サービスプロバイダがユーザーを識別し、課金できるようにする公開ＲＳＡ鍵のレジストリ(registry)から得られる。

Ｋ．保証された秘密鍵の安全性によるレジストリ

業者にエンドユーザーの識別を別個に独立して確認させるには、サービスプロバイダはモジュール１０の発行者の氏名、住所および他の識別情報と共に、特定モジュール１０の公開鍵を含むレジストリを維持したいと考えることがある。この目的のために、レジストリ内の公開鍵がモジュール１０にしかわからない秘密鍵に対応していることをサービスプロバイダが確実にすることが重要である。これを保証するために公開鍵がモジュール１０から抽出され、レジストリに入れられる際に、モジュール１０はサービスプロバイダが所有していなければならない。この情報をレジストリに記録した後、サービスプロバイダはレジストリに名前のあるエンドユーザーにモジュール１０を出荷できる。

モジュール１０をエンドユーザーが受け取った際に、秘密鍵がサービスプロバイダには知られず、またサービスプロバイダの被雇用者の誰にも知られていないことをエンドユーザーが確認できることも重要である。理想的なレジストリシステムはいずれかの当事者が他の当事者を信頼することを求めるべきでないので、このようなことは重要である。このシステムは各当事者が他の当事者のいずれも秘密鍵を知る可能性がないと確信できる時に限って、だれもが満足できるように働く。

これを達成するための１つの方法として、サービスプロバイダがモジュール１０にコマンドを送り、乱数を使った完全なＲＳＡ鍵セットをモジュール１０に発

生させ、次に、指数のうちの１つを自動的に秘密にし、よってだれもが秘密鍵の値を発見できないようにする方法がある。この鍵セットはサービスプロバイダによってカン内にプログラムされた鍵セットと異なる特殊なタイプを有し、よってモジュール１０により直接ビジネスを行う者がモジュール１０だけにしか秘密鍵が知られていないと、自ら判断できる。

１．準備

サービスプロバイダはアプリケーションに対するパスワードで保護された取引グループ４０を作成し、モジュール１０が発生したグループ内にＲＳＡ鍵を作成する。鍵セットの発生をモジュラスおよび指数は自動的にロックされるが、第２の指数はモジュール１０のファームウェアによって自動的に秘密にされる。サービスプロバイダは次に入力オブジェクトからのデータを秘密鍵で暗号化する取引スクリプト４４を作成し、暗号化された結果を出力オブジェクトに入れる。この取引スクリプト４４はオプションとして入力オブジェクトからのデータに別の情報（例えば取引カウンタ）を添付し、アプリケーションの別の目的を満たしてもよい。別のオブジェクト４２および取引スクリプト４４をサービスプロバイダの裁量で加えてもよい。取引グループ４０は完全になった時にサービスプロバイダによってロックされる。次に、サービスプロバイダは取引グループ４０からＲＳＡモジュラスおよび公開指数を読み出し、これをエンドユーザーを識別する情報と共にレジストリに記録する。最後に、サービスプロバイダはモジュール１０をエンドユーザーに出荷し、その後、取引グループ４０にアクセスするのに使用できるパスワードをエンドユーザーに送る。

２．使用

業者がインターネットまたは他のネットワークを通してエンドユーザーの確実な識別を得たいとき、業者はデータのユニークなパケットを発生し、これをエンドユーザーに送信し、エンドユーザーはデータを入力オブジェクトに送り、モジュール１０が発生した暗号鍵で暗号化させる取引スクリプト４４を要求する。この結果生じた暗号化されたパケットは業者へ戻されるよう送信される。次に業者は、エンドユーザーに属す公開鍵を得るようにサービスプロバイダによって提供

されるデータベースにアクセスし、エンドユーザーの公開鍵を使って暗号化されたパケットを解読しようとする。解読に成功すれば、業者は遠隔地のネットワーク化されたロケーションでエンドユーザーのモジュール１０が物理的に存在することを証明した。遠隔地におけるエンドユーザーのモジュール１０が存在することを保証することにより、この識別によりデータパケットの内容、従ってエンドユーザーがリクエストできるパケットの内容によって表示される金融取引を有効にし、かつ正当化する。

ここに述べたモデルは金銭取引を実行する権限がサービスプロバイダによって維持されるレジストリから得られるモデルとなっている。従って、この情報は正確であり、モジュール１０内の秘密鍵をすべての当事者から安全にできることが重要である。各モジュール１０は自己のユニークなＲＳＡ鍵セットを有するので、このモデルではモジュール１０がサービスプロバイダによって維持されるレジストリと独立してマネーを表示する手段はない。その代わりに、レジストリおよびモジュール１０が秘密鍵と共にサインできる能力が他の当事者と離間してエンドユーザーを識別するための定義手段として働く。

Ｌ．取引量の課税

この使用の章は、サービスプロバイダがモジュール１０によって送金されるマネーの総額のあるパーセントであるサービス料金をエンドユーザーから集めようとするビジネスモデルに関する。このモデルは上記のＣ、Ｄ、ＥおよびＦ章に記載されたものと類似しているが、所定の日および時間に特定の取引スクリプト４４を終了できるデストラクタオブジェクトが加えられている。このモデルはモジュール１０から送られたすべてのマネーの総計の値を累積するように（適当な取引スクリプト４４と共に）プログラムされる別のマネーオブジェクトを使用することも必要とする。

１．準備

サービスプロバイダは上記ＤおよびＥ章に記載したように、マネーオブジェクト等を含む取引グループ４０を作成する。サービスプロバイダは更に取引量累積

器として働くように別のマネーオブジェクトも作成する。このサービスプロバイ

ダはＤおよびＥにおけるように、マネーを引き出したり、預金するための取引スクリプト４４も作成するが、モジュール１０にマネーを加えるための取引スクリプトが将来の所定時間に終了するようにセットされたデストラクタオブジェクトおよびマネーを引き出すための取引スクリプト４４が引出し額を取引量累積器として働くマネーオブジェクトに加えるための命令を含む点が異なる。サービスプロバイダはグループをロックし、モジュール１０をエンドユーザーに出荷する。

２．使用

エンドユーザーは上記ＤおよびＥ章に記載されているように、預金および引出しのためのモジュール１０を使用する。モジュール１０を使用している間、モジュール１０から消費したマネーのすべての累積総量は取引量累積器として働くマネーオブジェクトに累積される。制限時間が終了すると、エンドユーザーはこのモジュール１０にマネーを加えることはできないが、所望すれば何も残らなくなるまでマネーを引き出し続けることができる。次にエンドユーザーはモジュール１０をサービスプロバイダへ戻し、回収される。サービスプロバイダはマネーの残高を読み出し、更に取引量累積器内に記録された金額を読み出す。サービスプロバイダは取引量累積器内の金額のパーセントであるサービス料金をエンドユーザーに請求する。エンドユーザーがこのサービスを続けるよう、この額を支払いたい場合、取引グループ４０を破壊し、再構築し、エンドユーザーが戻した時のモジュール１０内に残っていたマネーの残高が取引グループ４０のうちのマネーオブジェクトに戻されるようにプログラムされる。エンドユーザーがサービス料金を払うことを条件に、サービスプロバイダは回収されたモジュールをエンドユーザーに戻す。

上記システムによりサービスプロバイダはエンドユーザーが行うどの金融取引もモニタしたり、またこれに加入することなくサービスに対する定期料金を集めることができる。この料金は取引量レジスタの内容によって決定される実際の使用に基づく。

モジュールと共に使用するためのファームウェアの定義例

オブジェクト

モジュールファームウェアによって取り込まれ、操作される最も原始的なデータ構造である。次の章で有効オブジェクトおよびそれらの定義のリストについて説明する。

グループ

オブジェクトの内蔵された集合である。オブジェクトの範囲はメンバーであるグループに限定される。

グループＩＤ

特定グループを表示する、好ましくは０～２５５の間の番号。

オブジェクトＩＤ

特定グループ内の特定のオブジェクトを表示する、好ましくは０～２５５の間の番号。

オブジェクトタイプ

特定オブジェクトを記述する、好ましくは１バイトタイプの特定子(specifier)。

ＰＩＮ

好ましくは長さが８バイトの英数字個人識別番号

共通ＰＩＮ

監査トレイルのような共用リソースへのアクセスを制御するＰＩＮである。グループを作成し、削除するホストの能力を制御することにも使用される。

グループＰＩＮ

グループ内のオブジェクトに固有の操作に対するアクセスを制御するＰＩＮである。

監査トレイル

モジュールをロックした後に生じる取引の記録

ロックされたオブジェクト

オブジェクトロックコマンドを実行することによってロックされたオブジェクトのこと。一旦オブジェクトがロックされると、これを直接読み出すことはできない。

秘密オブジェクト

オブジェクト秘密化コマンドを実行することによって秘密にされたオブジェクト。オブジェクトが一旦秘密にされると、これを直接読み出したり書き込んだりすることはできなくなる。

ロックされたグループ

グループロックコマンドを使ってロックされたグループのこと。グループが一旦ロックされると、このグループはオブジェクトを作成できなくなる。

複合オブジェクト

いくつかのオブジェクトの組み合わせ。個々のオブジェクトは複合オブジェクトの属性を受け継ぐ。

オブジェクト定義の例

ＲＳＡモジュラス

長さが最大１２４ビットの大きな整数。これは、それぞれが所望するモジュールサイズの長さのビット数の約半分である２つの大きい素数の積である。このＲＳＡモジュラスはメッセージMを符号化したり解読したりするための次の式で使用される。

（１）暗号化：$C = M^e \pmod N$

（２）解読：$M = C^d \pmod N$

ここでCは暗号文であり、dおよびeはＲＳＡ指数（下記参照）であり、NはＲＳＡモジュラスである。

ＲＳＡ指数

（上記式１および２に示された）eおよびdの双方はＲＳＡ指数である。これらは一般に大きい数字であるが、モジュラス（N）よりも小さい。ＲＳＡ指数は秘密でも公開されていてもよい。モジュールでＲＳＡ指数を作成すると、これらは公開指数または秘密指数のいずれかに宣言できる。一旦作成されると、指数は公開指数から秘密指数に変更できる。しかしながら指数が秘密にされると、指数が属する取引グループ４０が破壊されるまで、この指数は秘密指数のままである。

取引スクリプト

取引スクリプトはモジュールが実行すべき一連の命令である。要求されるとモ

ジュールファームウェアはスクリプト内の命令を解読し、その結果を出力データ

オブジェクトに入れる（下記参照）。実際のスクリプトは単なるオブジェクトのリストにすぎない。オブジェクトがリストされている順序はオブジェクトで実行すべき操作を指定する。取引スクリプト４４は、好ましくは１２８バイトの長さとすることができる。

取引カウンタ

取引カウンタオブジェクトは長さが４バイトであることが好ましく、通常、作成時にゼロに初期化される。このオブジェクトを基準とする取引スクリプトが要求されるごとに、取引カウンタは１だけインクリメントされる。取引カウンタがロックされると、このカウンタが読み出され、取り消し不能なカウンタとなる。

マネーレジスタ

マネーレジスタオブジェクトは長さが４バイトであることが好ましく、マネーまたはその他の形態のクレジットを表示するのに使用できる。このオブジェクトが作成されると、ユーザーがこの値を不正に操作されるのを防止するようにロックしなければならない。一旦ロックされると取引スクリプトを要求することによってしかこのオブジェクトの値を変更できない。マネー取引を行う代表的な取引グループ４０は、マネーレジスタからの引き出しのためのスクリプトとマネーレジスタへの預け入れのためのスクリプトを有することができる。

クロックオフセット

このオブジェクトはモジュールのリアルタイムのクロックの表示とある任意の時間（例えば１９７０年１月１日１２：００ａｍ）との間の差を含む４バイトの数字であることが好ましい。この真の時間はクロックオフセットの値をリアルタイムクロックに加えることによりモジュールから得ることができる。

ＳＡＬＴ

ＳＡＬＴオブジェクトは長さが２０バイトであることが好ましく、作成時にランダムデータで初期化すべきである。ホストが発生ランダムＳＡＬＴコマンドを送信すると、モジュールは好ましくは新しいランダムＳＡＬＴを発生するように、先のＳＡＬＴと（好ましくはランダムに生じたパワーアップによって生じた）

モジュールの乱数とを組み合わせる。ＳＡＬＴオブジェクトが秘密にされない場合、このオブジェクトはその後、読み出しオブジェクトコマンドを発生することによ

り読み出しできる。

コンフィギュレーションデータ

このデータは好ましくは１２８バイトの最大長さを有する、ユーザーが定める構造である。このオブジェクトは一般に取引グループ４０に固有なコンフィギュレーション情報を記憶するのに使用される。例えば、このコンフィギュレーションデータオブジェクトを使ってマネーレジスタオブジェクトのフォーマット（すなわちオブジェクトが表示する通貨のタイプ）を指定できる。このオブジェクトは予め定義された構造を有しないので、取引オブジェクトによって使用することはできない。

入力データ

入力データオブジェクトは、好ましくは１２８バイトの最大長さを有する入力バッファにすぎない。１つの取引グループは多数の入力オブジェクトを有することができる。コストは入力データオブジェクトを使って取引スクリプト４４によって処理すべきデータを記憶する。

出力データ

この出力データオブジェクトは出力バッファとして取引スクリプトによって使用される。取引グループが作成される際に、このオブジェクトも自動的に作成される。このオブジェクトは好ましくは長さが５１２バイトであり、そのグループからパスワードの保護を引き継ぐ。

ランダムフィル

スクリプトインタープリタがこのタイプのオブジェクトに出会うと、カレントメッセージを自動的にパッドし、その長さを先行するモジュラスの長さよりも１ビット短くする。取引グループが作成される際に、このオブジェクトに対するハンドルも自動的に作成される。このオブジェクトは秘密オブジェクトであり、読み出しオブジェクトコマンドを使って読み出すことはできない。

ワーキングレジスタ

このオブジェクトはワーキングレジスタスペースとしてスクリプトインタープリタによって使用され、取引スクリプト内で使用できる。取引グループが作成される際に、このオブジェクトに対するハンドルも自動的に作成される。このオブジェクトは秘密オブジェクトであり、読み出しオブジェクトコマンドを使って読み出しできない。

ＲＯＭデータ

このオブジェクトは取引グループが作成される際に自動的に作成される。このオブジェクトはロックされたオブジェクトであり、書き込みオブジェクトコマンドを使って変更できない。このオブジェクトは長さが８バイトであり、その内容はマイクローインーＡーカン（商標）のＲＯＭデータの８倍と同じである。

好ましいモジュールファームウェアコマンドのセット

共通ＰＩＮセットコマンド（０１Ｈ）

（モジュールへの）送信

０１Ｈ、旧ＰＩＮ、新ＰＩＮ、ＰＩＮオプションバイト

受信データ

ＣＳＢ（コマンドステータスバイト）＝成功した場合は０であり、

そうでなかった場合は適当なエラーコード

出力長さ＝０

出力データ＝０

注：ＰＩＮオプションバイトはビット状でもよいし、次の値のいずれかでよい。

| | |
|---|---|
| ＰＩＮ＿ＴＯ＿ＥＲＡＳＥ | ００００００００１ｂ（マスター消去に対してはＰＩＮを必要とする） |
| ＰＩＮ＿ＴＯ＿ＣＲＥＡＴＥ | ００００００１０ｂ（グループ作成に対してはＰＩＮを必要とする） |

最初、モジュールは０（ゼロ）のＰＩＮ（個人識別番号）および０のオプションバイトを有する。一旦ＰＩＮが設定されると、旧ＰＩＮを与えるか、またはマ

スター消去によってしか変更できない。しかしながらオプションバイト内にＰＩＮ＿ＴＯ＿ＥＲＡＳＥがセットされていれば、このＰＩＮは共通ＰＩＮセットコマンドを用いた場合に限り変更できる。

共通ＰＩＮセットコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮ　（共通ＰＩＮの一致なし）

ＥＲＲ＿ＢＡＤ＿ＰＩＮ＿ＬＥＮＧＴＨ　（新ＰＩＮ長さ＞８バイト）

ＥＲＲ＿ＢＡＤ＿ＯＰＴＩＯＮ＿ＢＹＴＥ　（認識不能なオプションバイト）

この章で説明するすべてのコマンドに対し、ホストにより受信されるデータはリターンパケット状となる。リターンパケットは次の構造を有する。

コマンドステータスバイト　（コマンドが成功した場合、０、そうでないエラーコード（１バイト））

出力データ長さ　（コマンド出力長さ、２バイト）

出力データ　（コマンド出力、上記長さ）

マスター消去コマンド（０２Ｈ）

送信データ

０２Ｈ、共通ＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０で、成功しなかった場合ＥＲＲＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮとなる。

出力長さ＝０

出力データ＝０

注：ＰＩＮオプションのＬＳＢ（最小位ビット）がクリア（すなわちマスター消去に対しＰＩＮが不要）である場合、共通ＰＩＮ値に対し０を送信する。一般に本明細書では常にＰＩＮが必要であると見なす。ＰＩＮが設定されていなければ、ＰＩＮとして０を送信すべきである。これは共通ＰＩＮおよびグループＰＩＮにも当てはまる（下記参照）。ＰＩＮが正しければ、ファームウェアはすべて

のグループ（下記参照）およびこれらグループ内のすべてのオブジェクトを削除する。共通ＰＩＮおよび共通ＰＩＮオプションバイトの双方を０にリセットする。

すべてを消去した後にモジュールはリターンパケットを送信する。ＣＳＢは既に述べたとおりである。出力データ長さおよび出力データフィールドの双方を０にセットする。

グループ作成コマンド（０３Ｈ）

送信データ

０３Ｈ、共通ＰＩＮ、グループ名称、グループＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

成功した場合、出力長さ＝１であり、

成功しなかった場合、０であり、

成功した場合、出力データ＝グループＩＤであり、成功しなかった場合０となる。

注：最大グループ名称の長さは１６バイトであり、最大ＰＩＮ長さは８バイトである。ＰＩＮ＿ＴＯ＿ＣＲＥＡＴＥビットが共通ＰＩＮオプションバイト内でセットされ、送信されたＰＩＮが共通ＰＩＮに一致しない場合、モジュールはＯＳＣをＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮにセットする。

グループ作成コマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮ　（不正確な共通ＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＮＡＭＥ＿ＬＥＮＧＴＨ　（グループ名称の長さ＞１６バイトの場合）

ＥＲＲ＿ＢＡＤ＿ＰＩＮ＿ＬＥＮＧＴＨ　（グループＰＩＮ長さ＞８バイトの場合）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

ＥＲＲ＿ＩＮＳＵＦＦＩＣＩＥＮＴ＿ＲＡＭ　（新しいグループに対し、メモリは充分でない）

グループＰＩＮセットコマンド（０４Ｈ）

送信データ

０４Ｈ、グループＩＤ、旧ＧＰＩＮ、新ＧＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

出力長さ＝０

出力データ＝０

注：グループＰＩＮはコマンドパケット内で送信されたグループＩＤにより指定されたグループ内のオブジェクトへのアクセスを限定するだけである。

セットグループＰＩＮコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（グループのＰＩＮは一致しない）

ＥＲＲ＿ＢＡＤ＿ＰＩＮ＿ＬＥＮＧＴＨ　（新グループＰＩＮ長さ＞８バイト）

オブジェクト作成コマンド（０５Ｈ）

送信データ

０５Ｈ、グループＩＤ、グループＰＩＮ、オブジェクトタイプ、オブジェクト属性、オブジェクトデータ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

成功した場合、出力長さ＝１であり、

成功しなかった場合、０であり、

成功した場合、出力データ＝オブジェクトＩＤであり、成功しなかった場合では０となる。

注：オブジェクト作成コマンドが成功した場合、モジュールファームウェアはグループＩＤによって指定されたグループ内のオブジェクトＩＤを戻す。ホストによって供給されるＰＩＮが不正確であるか、またはグループがグループロックコマンドによってロックされている場合（下記記載）、モジュールはＣＳＢ内のエラーコードを戻す。何らかの理由からオブジェクトが無効の場合、オブジェクトの作成もできない。例えば作成中のオブジェクトがＲＳＡモジュラス（タイプ０）であり、長さが１０２４ビットより大きい場合。すべての取引スクリプトの規則に従う場合、取引スクリプトの作成に成功する。

オブジェクト作成コマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤ　（グループは既にロックされている）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

ＥＲＲ＿ＩＮＶＡＬＩＤ＿ＴＹＰＥ　（指定されたオブジェクトタイプは無効である）

ＥＲＲ＿ＢＡＤ＿ＳＩＺＥ　（オブジェクト長さは無効である）

ＥＲＲ＿ＩＮＳＵＦＦＩＣＩＥＮＴ＿ＲＡＭ　（新しいオブジェクトに対し、メモリは充分でない）

| | | |
|---|---|---|
| オブジェクトタイプ：ＲＳＡモジュール | | 0 |
| | ＲＳＡ指数 | 1 |
| | マネーレジスタ | 2 |
| | 取引カウンタ | 3 |
| | 取引スクリプト | 4 |
| | クロックオフセット | 5 |
| | ランダムＳＡＬＴ | 6 |
| | コンフィギュレーションオブジェクト | 7 |
| | 入力データオブジェクト | 8 |

出力データオブジェクト　　9

オブジェクト属性：　ロックされた状態　０００００００１ｂ

秘密にされた状態　０００００００１０ｂ

下記のオブジェクトロックコマンドおよびオブジェクト秘密化コマンドを使用することにより、作成後にオブジェクトをロックし、秘密にすることもできる。

オブジェクトロックコマンド（０６Ｈ）

送信データ

０６Ｈ、グループＩＤ、グループＰＩＮ、オブジェクトＩＤ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

出力長さ＝０

出力データ＝０

注：グループＩＤ、グループＰＩＮおよびオブジェクトＩＤのすべてが正しい場合、モジュールは指定されたオブジェクトをロックする。オブジェクトをロックすることは取り消し不能な操作である。

オブジェクトロックコマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤ　（グループは既にロックされている）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（指定されたオブジェクトは存在しない）

オブジェクト秘密化コマンド（０７Ｈ）

送信データ

０７Ｈ、グループＩＤ、グループＰＩＮ、オブジェクトＩＤ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合では適当なエラーコードとなる。

注：グループＩＤ、グループＰＩＮ、オブジェクトＩＤが有効な場合、オブジェクトは秘密にされる。秘密にされたオブジェクトはロックされたオブジェクトのすべての性質を共用するが、読み取りできない。秘密にされたオブジェクトは取引スクリプトによってしか変更できない。秘密にされたオブジェクトをロックすることは適法であるが、オブジェクトの秘密化はオブジェクトのロッキングよりも強力な操作であるので、このことには意味がない。オブジェクトを秘密化することは取り消し不能な操作である。

オブジェクト秘密化コマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤ　（グループは既にロックされている）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（指定されたオブジェクトは存在しない）

オブジェクトを破壊可能にするコマンド（０８Ｈ）

送信データ

０８Ｈ、グループＩＤ、グループＰＩＮ、オブジェクトＩＤ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合では適当なエラーコードとなる。

注：グループＩＤ、グループＰＩＮ、オブジェクトＩＤが有効な場合、オブジ

ェクトは破壊可能にされる。オブジェクトが破壊可能にされると、グループデストラクタがアクティブになった後にはこのオブジェクトは取引スクリプトによって使用不能となる。取引グループ内に破壊オブジェクトが操作しない場合、破壊可能なオブジェクトの属性ビットは影響がない。オブジェクトを破壊可能にすることは取り消し不能な操作である。

オブジェクトを破壊可能にするコマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤ　（グループは既にロックされている）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（指定されたオブジェクトは存在しない）

モジュールロックコマンド（０９Ｈ）

送信データ

０９Ｈ、共通ＰＩＮ

受信データ

コマンドが成功すればＣＳＢ＝０であり、成功しなかった場合、適当なエラーコード

成功した場合、出力長さ＝２であり、そうでなかった場合、０

成功した場合、出力データ＝監査トレイルサイズであり、そうでない場合、０である。

注：ホストの供給した共通ＰＩＮが正しく、モジュールが先にロックされている場合、コマンドは成功する。モジュールがロックされていると、モジュールは新しいグループまたはオブジェクトを受け入れない。このことは、すべてのグル

ープが自動的にロックされることを意味する。システムまたはグループによって使用されないＲＡＭは監査トレイルのために使用される。モジュールが成功裏にロックされるまで、監査トレイルはない。

監査トレイルの記録は６バイト長さであり、次の構造を有する。

グループＩＤ｜オブジェクトＩＤ｜日／時間スタンプ

一旦監査トレイルが設定されると、取引スクリプトが実行される毎に、利用可能なサイズのバイトの第１ロケーション内に、上に示されたフォームの記録が記憶される。ここで、監査トレイルが開始する前にモジュールはロックされなければならないので、グループＩＤもオブジェクトＩＤも変更を受けないことに留意されたい。これにより常に監査トレイルを処理するアプリケーションは実行された取引スクリプトをユニークに識別できる。監査トレイルが利用可能なメモリのすべてを一旦消費すると、監査トレイルは最も古い取引記録の上に新しい取引記録を記憶する。

モジュールロックコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮ　（供給される共通ＰＩＮは不正確）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールは既にロックされている）

グループロックコマンド（０ＡＨ）

送信データ

０ＡＨ、グループＩＤ、グループＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合、適当なエラーコードとなる。

出力長さ＝０

出力データ＝０

注：提供されるグループＰＩＮが正しい場合、モジュールＢＩＯＳは特定されたグループ内に更にオブジェクトが作成されるのを許可しない。グループは完全内

蔵エンティティであるので、（後述する）グループ削除コマンドを実行することにより、これらを削除してもよい。

ロックグループコマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤ　（グループは既にロックされている）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（指定されたオブジェクトは存在しない）

取引スクリプト要求コマンド（０ＢＨ）

送信データ

０ＢＨ、グループＩＤ、グループＰＩＮ、オブジェクトＩＤ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合では適当なエラーコードとなる。

成功した場合、出力長さ＝１であり、そうでない場合０

出力データ＝予想される完了時間

注：モジュールによって戻された予想時間は１６分の１秒である。ＣＳＢ内にエラーコードが戻されると、予想時間は０となる。

取引スクリプト実行コマンドに対する可能性のあるエラーリターンコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（指定されたオブジェクトは存在しない）

オブジェクト読み出しコマンド（０ＣＨ）

送信データ

０ＣＨ、グループＩＤ、グループＰＩＮ、オブジェクトＩＤ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

成功した場合、出力長さ＝オブジェクト長さであり、そうでない場合０

成功した場合、出力データ＝オブジェクトデータであり、そうでない場合、０

注：グループＩＤ，グループＰＩＮおよびオブジェクトＩＤが正しければ、モジュールは指定されたオブジェクトの属性バイトをチェックする。オブジェクトが秘密化されていなければ、モジュールはオブジェクトデータをホストに送信する。グループＰＩＮが無効か、オブジェクトが秘密化されていなければ、モジュールは出力長さ内の０およびリターンパケットのデータフィールドを戻す。

オブジェクト読み出しコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（指定されたオブジェクトは存在しない）

ＥＲＲ＿ＯＢＪＥＣＴ＿ＰＲＩＶＡＴＩＺＥＤ（オブジェクトは秘密化されている）

オブジェクト書き込みコマンド（０ＤＨ）

送信データ

０ＤＨ、グループＩＤ、グループＰＩＮ、オブジェクトＩＤ、オブジェクトサイズ、オブジェクトデータ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合では

適当なエラーコードとなる。

出力長さ＝０

出力データ＝０

注：グループＩＤ、グループＰＩＮおよびオブジェクトＩＤのすべてが正しい場合、モジュールは指定されたオブジェクトの属性バイトをチェックする。オブジェクトがロックされているか、秘密にされている場合、モジュールはオブジェクトの先のサイズおよびデータをクリアし、これを新しいオブジェクトデータを取り替える。オブジェクトのタイプおよび属性バイトは影響されないことに留意。

オブジェクト書き込みコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ　（オブジェクトはグループ内に存在していなかった）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＳＩＺＥ　（指定された違法なオブジェクトサイズ）

ＥＲＲ＿ＯＢＪＥＣＴ＿ＬＯＣＫＥＤ　（オブジェクトはロックされている）

ＥＲＲ＿ＯＢＪＥＣＴ＿ＰＲＩＶＡＴＩＺＥＤ　（オブジェクトは秘密化されている）

グループ名称読み出しコマンド（０ＥＨ）

送信データ

０ＥＨ、グループＩＤ

受信データ

ＣＳＢ＝０

出力長さ＝グループ名称の長さ

出力データ＝グループ名称

注：グループ名称は最大１６バイト出あり、グループ名称内ですべてのバイト値は適法である。

グループ削除コマンド（０ＦＨ）

送信データ

０ＤＨ、グループＩＤ、グループＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合、適当なエラーコードとなる。

出力長さ＝０

出力データ＝０

注：グループＰＩＮおよびオブジェクトＩＤが正しい場合、モジュールは指定されたグループを削除する。グループの削除によりグループ内のすべてのオブジェクトが自動的に削除される。モジュールがすでにロックされている場合、グループ削除コマンドは成功しない。

グループ削除コマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（不正確なグループＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在していない）

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ　（モジュールはロックされている）

コマンドステータス情報入手コマンド（１０Ｈ）

送信データ

１０Ｈ

受信データ

ＣＳＢ＝０

出力長さ＝６

出力データ＝モジュールステータス構造（下記参照）

注：この操作はＰＩＮを必要をせず、決して失敗しない。ステータス構造はつ

ぎのように定義される。

最後に実行されたコマンド（１バイト）

最終コマンドステータス（１バイト）

受信した時間コマンド（４バイト）

モジュールコンフィギュレイション情報入手コマンド（１１Ｈ）

送信データ

１１Ｈ

受信データ

ＣＳＢ＝０

出力長さ＝４

出力データ＝モジュールコンフィギュレイション構造

注：この操作はＰＩＮを必要をせず、決して失敗しない。ステータス構造はつぎのように定義される。

グループ数（１バイト）

フラグバイト（下記参照）（１バイト）

監査トレイルサイズ／自由ＲＡＭ（２バイト）

このフラグのタイプはビット状であるか、つぎの値のいずれかである。

００００００１ｂ（モジュールはロックされている）

００００００１０ｂ（アクセスに必要な共通ＰＩＮ）

監査トレイル情報読み出しコマンド（１２Ｈ）

送信データ

１２Ｈ、共通ＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、そうでない場合、適当なエラーコードとなる。

成功した場合、出力長さ＝新しい記録の＃×６、そうでない場合、０

出力データ＝新しい監査トレイル記録

注：送信された共通ＰＩＮが有効であり、モジュールがロックされている場合

、新しいすべての取引記録はホストに転送される。

監査トレイル情報読み出しコマンドに対する可能性のあるエラーコード

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮ　（共通ＰＩＮは正しくない）

ＥＲＲ＿ＭＩＡＣ＿ＮＯＴ＿ＬＯＣＫＥＤ　（モジュールはロックされていない）

監査トレイル読み出しコマンド（１３Ｈ）

送信データ

１３Ｈ、共通ＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

成功した場合、新しい記録の数×６で、そうでない場合、０

出力データ＝新しい監査トレイルの記録

注：送信されたＰＩＮが有効でモジュールがロックされている場合、ＰＩＮは新しい取引の記録のすべてをホストへ転送する。

監査トレイル読み出しコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮ　（共通ＰＩＮは不正確）

ＥＲＲ＿ＭＩＡＣ＿ＮＯＴ＿ＬＯＣＫＥＤ　（モジュールはロックされていない）

グループ監査トレイル読み出しコマンド（１４Ｈ）

送信データ

１４Ｈ、グループＩＤ、グループＰＩＮ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

成功した場合、出力長さ＝グループに対する記録の数×６で、そうでない場合、０

出力データ＝グループに対する監査トレイルの記録

注：このコマンドは監査トレイル読み出しコマンドと同じであるが、送信データ内で特定されたグループＩＤに関連した記録しかホストに戻されないことが異なっている。これにより、他のグループの記録を見ることなく取引グループが自己の活動を記録しトラックできる。

グループ監査トレイル読み出しコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（グループＩＤは存在せず）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（共通ＰＩＮは正しくない）

ＥＲＲ＿ＭＩＡＣ＿ＮＯＴ＿ＬＯＣＫＥＤ　（モジュールはロックされていない）

リアルタイムクロック読み出しコマンド（１５Ｈ）

送信データ

１５Ｈ、共通ＰＩＮ

受信データ

共通ＰＩＮが一致する場合、ＣＳＢ＝０であり、そうでない場合ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮとなる。

出力長さ＝４

出力データ＝リアルタイムクロックの最大４つの位のバイト

注：この値はクロックオフセットでは調節されない。このコマンドは取引グループ作成中にクロックオフセットを計算するようサービスプロバイダよって通常使用される。

調節されたリアルタイムクロック読み出しコマンド（１６Ｈ）

送信データ

１６Ｈ、グループＩＤ，グループＰＩＮ、オフセットオブジェクトのＩＤ

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、成功しなかった場合適当なエラーコードとなる。

成功した場合、出力長さ＝４、そうでない場合、０

出力データ＝リアルタイムクロック＋クロックオフセットＩＤ

注：グループＩＤ及びグループＰＩＮが有効であり、オブジェクトＩＤがクロックオフセットのＩＤである場合、このコマンドは成功する。このモジュールはＲＴＣの最大位の４バイトの現在の値にクロックオフセットを加える。同じタスクを実行し、その結果を出力データオブジェクトに入れるよう取引スクリプトを書き込むことができることに留意のこと。

調節されたリアルタイムクロック読み出しコマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（正確でないグループＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＴＹＰＥ　（オブジェクトＩＤはクロックオフセットでない）

ランダムデータを得るコマンド（１７Ｈ）

送信データ

１７Ｈ、長さ（Ｌ）

受信データ

コマンドが成功した場合、ＣＳＢ＝０であり、そうでない場合エラーコード。

成功した場合、出力長さ＝Ｌ、そうでない場合は０

成功した場合、出力データ＝ランダムデータのＬバイト

注：このコマンドは暗号学的に有効な乱数の良好なソースとなる。

ランダムデータを得るコマンドの可能性のあるコード

ＥＲＲ＿ＢＡＤ＿ＳＩＺＥ　（リクエストされたバイト数＞１２８）

ファームウェアのバージョンＩＤを得るコマンド（１８Ｈ）

送信データ

１８Ｈ

受信データ

ＣＳＢ＝０

出力長さ＝ファームウェアのバージョンＩＤのストリングの長さ

出力データ＝ファームウェアのバージョンＩＤのストリング

注：このコマンドはパスカルタイプのストリング（長さ＋データ）としてファームウェアバージョンＩＤを戻す。

フリーＲＡＭを得るコマンド（１９Ｈ）

送信データ

１９Ｈ

受信データ

ＣＳＢ＝０

出力長さ＝２

出力データ＝フリーＲＡＭの量を含む２バイト値

注：モジュールがロックされている場合、出力データバイトはいずれも０となり、取引グループによって使用されないすべてのメモリが監査トレイルに対して留保されていることを示す。

グループ名称変更コマンド（１ＡＨ）

送信データ

１ＡＨ、グループＩＤ、グループＰＩＮ、新しいグループ名称

受信データ

コマンドが成功した場合、ＣＳＢ＝０、そうでない場合、適当なエラーコード

出力長さ＝０

出力データ＝０

注：指定されたグループＩＤがモジュール内に存在し、供給されたＰＩＮが正しい場合、取引グループ名称はホストによって供給された新しいグループ名称と置き換えられる。０のグループＩＤが供給される場合、送信されるＰＩＮは共通ＰＩＮでなければならない。正しい場合、モジュール名はホストによって供給される新しい名称と置換される。

グループ名称変更コマンドに対する可能性のあるエラーコード：

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ　（正確でないグループＰＩＮ）

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ　（指定されたグループは存在しない）

ＥＲＲ＿ＢＡＤ＿ＮＡＭＥ＿ＬＥＮＧＴＨ　（新しいグループ名称＞１６バイト）

エラーコードの定義

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＡＮＤ（８０Ｈ）

このエラーコードはモジュールファームウェアがホストの送信した直後のコマンドを認識しない時に発生する。

ＥＲＲ＿ＢＡＤ＿ＣＯＭＭＯＮ＿ＰＩＮ（８１Ｈ）

このエラーコードはコマンドが共通ＰＩＮを求め、供給されたＰＩＮがモジュールの共通ＰＩＮに一致しない時に戻される。最初、この共通ＰＩＮは０にセットされる。

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮ（８２Ｈ）

取引グループは自己のＰＩＮを有することができる（図１１）。このＰＩＮが（グループＰＩＮセットコマンドによって）セットされている場合、このＰＩＮはグループ内のオブジェクトのいずれかにアクセスするように供給されなければならない。供給されたグループＰＩＮが実際のグループＰＩＮに一致しない場合、モジュールはＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＰＩＮエラーコードを戻す。

ＥＲＲ＿ＢＡＤ＿ＰＩＮ＿ＬＥＮＧＴＨ（８３Ｈ）

ＰＩＮ値を変更できるコマンドは２つある。すなわちグループＰＩＮセットコマンドと、共通ＰＩＮセットコマンドである。これらの双方は新しいＰＩＮだけでなく旧ＰＩＮも必要とする。供給された旧ＰＩＮが正しいが、新ＰＩＮの長さが８キャラクタよりも長い場合、このＥＲＲ＿ＢＡＤ＿ＰＩＮ＿ＬＥＮＧＴＨエラーコードは戻される。

ＥＲＲ＿ＢＡＤ＿ＯＰＴＩＯＮ＿ＢＹＴＥ（８４Ｈ）

このオプションバイトは共通ＰＩＮにしか使用されない。共通ＰＩＮセットコマンドが実行されるとホストが供給する最終バイトは（コマンドの章に記載した）オプションバイトとなる。このバイトがモジュールに認識不能である場合、

このバイトはＥＲＲ＿ＢＡＤ＿ＯＰＴＩＯＮ＿ＢＹＴＥエラーコードを戻す。

ＥＲＲ＿ＢＡＤ＿ＮＡＭＥ＿ＬＥＮＧＴＨ（８５Ｈ）

取引グループ作成コマンドが実行されると、ホストによって供給されるデータ構造の１つはグループ名称となる。このグループ名称は長さが１６キャラクタを越えることはできない。供給される名称が１６キャラクタよりも長い場合、ＥＲＲ＿ＢＡＤ＿ＮＡＭＥ＿ＬＥＮＧＴＨエラーコードが戻される。

ＥＲＲ＿ＩＮＳＵＦＦＩＣＩＥＮＴ＿ＲＡＭ（８６Ｈ）

モジュール内で利用できる充分な量がない場合、取引グループ作成コマンドおよびオブジェクト作成コマンドはこのエラーコードを戻す。

ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤ（８７Ｈ）

モジュールがロックされていると、どのグループも、またはオブジェクトも作成したり破壊したりすることはできない。オブジェクトを作成または削除する試みにより、ＥＲＲ＿ＭＩＡＣ＿ＬＯＣＫＥＤエラーコードが発生される。

ＥＲＲ＿ＭＩＡＣ＿ＮＯＴ＿ＬＯＣＫＥＤ（８８Ｈ）

モジュールがロックされていない場合、監査トレイルはない。監査トレイルコマンドのうちの１つが実行される場合、このエラーコードは戻される。

ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤ（８９Ｈ）

一旦取引グループがロックされると、そのグループ内でのオブジェクトの作成は不可能となる。オブジェクトの属性およびタイプも凍結される。オブジェクトを作成したり、それら属性またはバイトのタイプを変更する試みがあれば、ＥＲＲ＿ＧＲＯＵＰ＿ＬＯＣＫＥＤエラーコードが発生される。

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＴＹＰＥ（８ＡＨ）

ホストがオブジェクト作成コマンドをモジュールに送ると、コマンドが供給するパラメータのうちの１つはオブジェクトタイプとなる（コマンドの章参照）。オブジェクトタイプがファームウェアによって認識されない場合、このオブジェクトはＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＴＹＰＥエラーコードを戻す。

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＡＴＴＲ（８ＢＨ）

ホストがオブジェクト作成コマンドをモジュールに送ると、このコマンドが供給するパラメータのうちの１つはオブジェクト属性バイトとなる（コマンドの章

参照）。オブジェクト属性バイトがファームウェアによって認識されない場合、このコマンドはＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＡＴＴＲエラーコードを戻す。

ＥＲＲ＿ＢＡＤ＿ＳＩＺＥ（８ＣＨ）

オブジェクトを作成または書き込む際に、通常、ＥＲＲ＿ＢＡＤ＿ＳＩＺＥエラーコードが発生される。ホストによって供給されるオブジェクトデータが無効長さを有する場合に限り、このことが生じる。

ＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤ（８ＤＨ）

取引グループレベルで作動するすべてのコマンドは、グループＩＤがコマンドパケットで供給されることを求める。指定されたグループＩＤがモジュール内に存在しない場合、このＩＤはＥＲＲ＿ＢＡＤ＿ＧＲＯＵＰ＿ＩＤエラーコードを発生する。

ＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤ（８ＥＨ）

オブジェクトレベルで作動するすべてのコマンドはオブジェクトＩＤがコマンドパケットで供給されることを求める。指定されたオブジェクトＩＤが特定の取引グループ（これもコマンドパケット内で指定される）内に存在しない場合、このモジュールはＥＲＲ＿ＢＡＤ＿ＯＢＪＥＣＴ＿ＩＤエラーコードを発生する。

ＥＲＲ＿ＩＮＳＵＦＦＩＣＩＥＮＴ＿ＦＵＮＤＳ（８ＦＨ）

金融取引を行うスクリプトオブジェクトが要求され、マネーレジスタの値が求められた引き出し額よりも少ない場合、ＥＲＲ＿ＩＮＳＵＦＦＩＣＩＥＮＴ＿ＦＵＮＤＳエラーコードが戻される。

ＥＲＲ＿ＯＢＪＥＣＴ＿ＬＯＣＫＥＤ（９０Ｈ）

ロックされたオブジェクトが読み出されるだけである。オブジェクト書き込みコマンドが試みられ、このコマンドがロックされたオブジェクトのオブジェクトＩＤを指定する場合、モジュールはＥＲＲ＿ＯＢＪＥＣＴ＿ＬＯＣＫＥＤエラーコードを戻す。

ＥＲＲ＿ＯＢＪＥＣＴ＿ＰＲＩＶＡＴＥ（９１Ｈ）

秘密オブジェクトは直接読み出しできないし、書き込みもできない。オブジェクト読み出しコマンドまたはオブジェクト書き込みコマンドが試みられ、コマンドが秘密オブジェクトのオブジェクトＩＤを指定すると、コマンドはＥＲＲ＿Ｏ

ＢＪＥＣＴ＿ＰＲＩＶＡＴＥエラーコードを戻す。

ＥＲＲ＿ＯＢＪＥＣＴ＿ＤＥＳＴＲＵＣＴＥＤ（９２Ｈ）

オブジェクトが破壊可能であり、取引グループのデストラクタがアクティブである場合、スクリプトによりオブジェクトは使用できない。破壊されたオブジェクトを使用するスクリプトが要求された場合、モジュールによりＥＲＲ＿ＯＢＪＥＣＴ＿ＤＥＳＴＲＵＣＴＥＤエラーコードが戻される。

本発明の実施例は耐久性のあるステンレススチール製のトークン状の缶に入れることが好ましい。モジュールは実質的に連結可能な物品内に入れることができると解される。連結可能な物品の例としては、クレジットカード、リング、腕時計、がま口、財布、ネックレス、宝石、ＩＤバッジ、ペン、クリップボード等がある。

モジュールは単一チップの、信頼されたコンピュータとすることが好ましい。「信頼された」なる用語は、このコンピュータが好ましくない手段によって不正な扱いから極めて安全であることを意味する。このモジュールは数学的に集中される暗号化を行うよう最適にされた数値コプロセッサを含む。ＢＩＯＳは変更に対して耐力があり、極めて安全な取引ができるように特別に設計されていることが好ましい。

各モジュールは秘密鍵／公開鍵セットを作成する能力を備えた乱数シード発生器を有することができる。秘密鍵はモジュールから決して離れることなく、モジュールにしか知られていない。更に秘密鍵の発見はモジュールへの不正な侵入に対してアクティブな自動破壊作用によって防止される。このモジュールは個人の識別番号（ＰＩＮ）によってユーザーに対して拘束できる。

取引がモジュールによって実行されると、モジュールおよびモジュールが通信するシステムのいずれかまたは双方によって正当化の証明書が作成される。特にこの証明書は下記を含むことができる。

１）ユニークな登録番号により、モジュールユーザーであるのは誰か。

２）取引の真の時間スタンプにより、取引がいつ行われたか。

３）登録されたモジュールインターフェースのサイトの識別により、どこで取引が行われたか。

４）ユニークに連続にされた取引およびメッセージダイジェスト上のデジタル署名による安全情報。

５）有効、紛失または終了のように表示されたモジュールステータス。

以上で、本発明の方法および装置の好ましい実施例について添付図面に示し、これまでの詳細な説明に述べたが、本発明は開示した実施例のみに限定されるものでなく、次の請求の範囲に記載した発明の範囲から逸脱することなく、多数の再配置、変更および置換が可能であると理解できよう。

【図１】

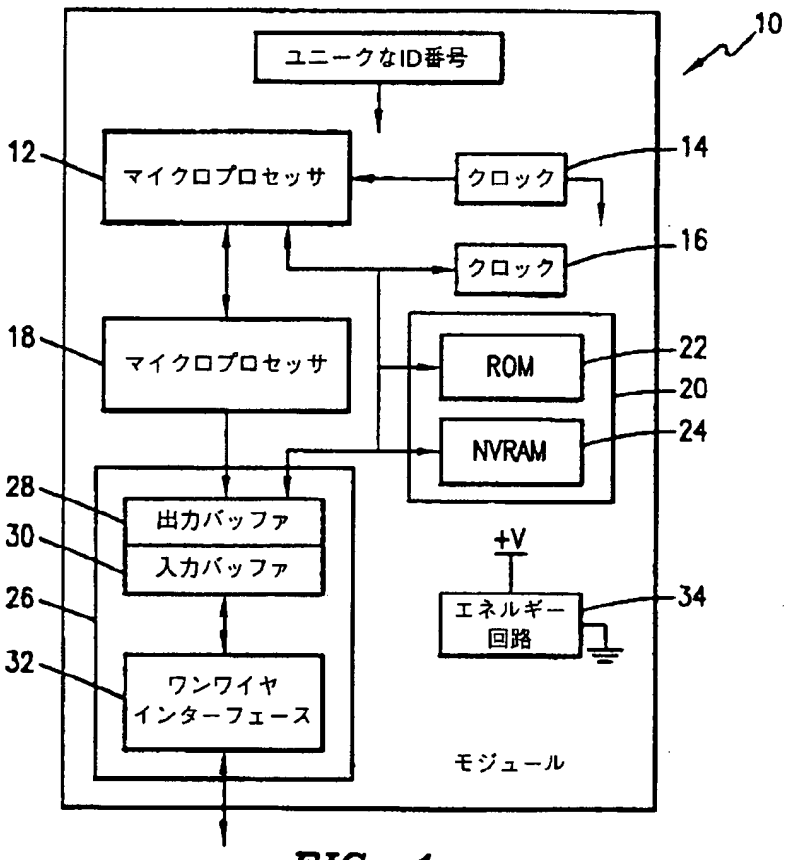



FIG. 1

【図２】

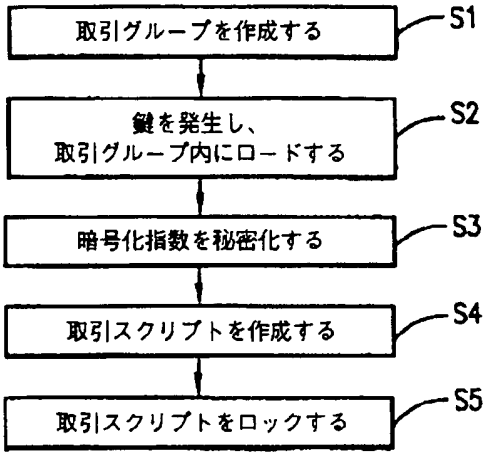
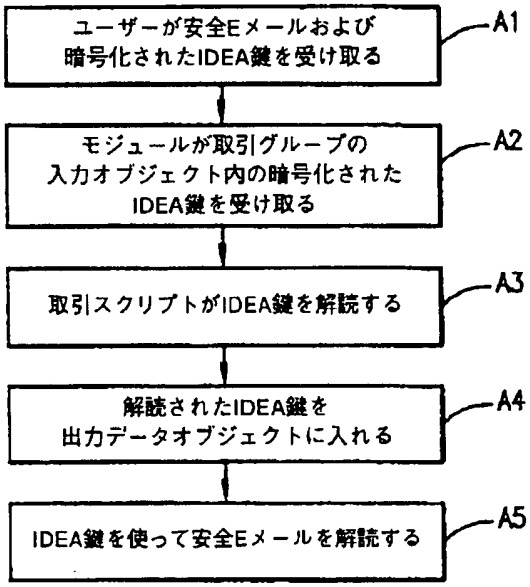



*FIG. 2*

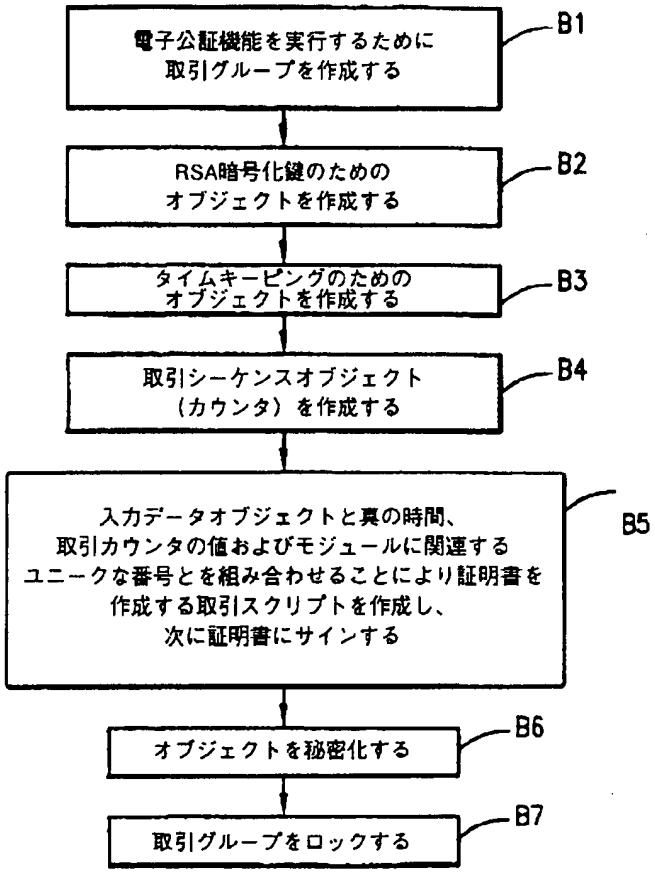
【図３】

*FIG. 3*



【図４】

*FIG. 4*


電子公証機能を実行するために
取引グループを作成する
B1
RSA暗号化鍵のための
オブジェクトを作成する
B2
タイムキーピングのための
オブジェクトを作成する
B3
取引シーケンスオブジェクト
（カウンタ）を作成する
B4
入力データオブジェクトと真の時間、
取引カウンタの値およびモジュールに関連する
ユニークな番号とを組み合わせることにより証明書を
作成する取引スクリプトを作成し、
次に証明書にサインする
B5
オブジェクトを秘密化する
B6
取引グループをロックする
B7

【図５】

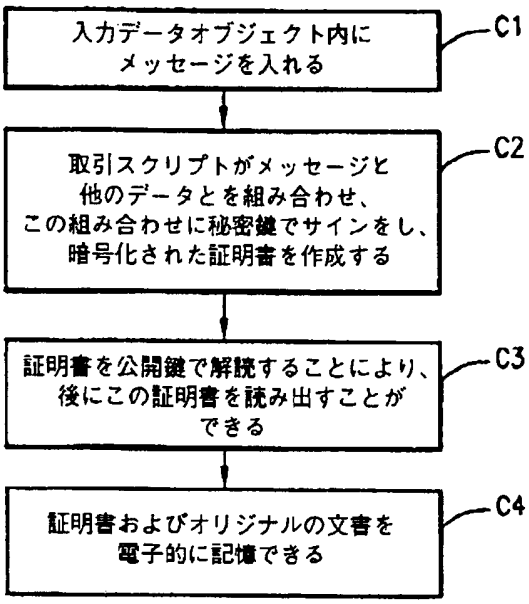



*FIG. 5*

【図６】

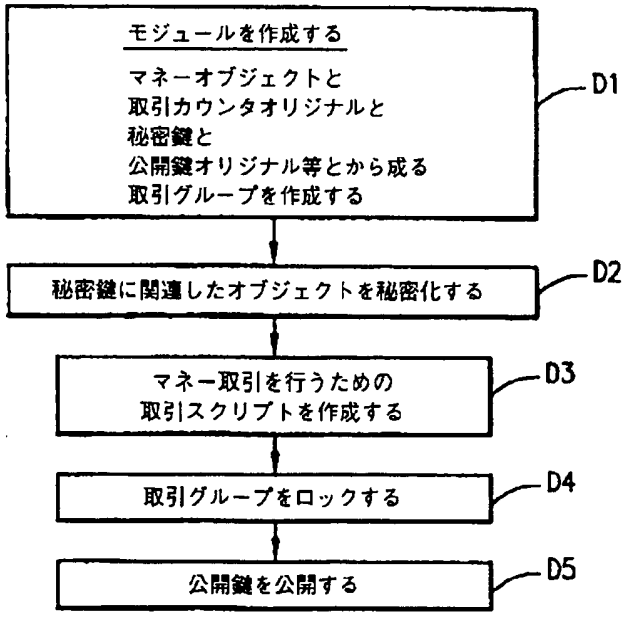



*FIG. 6*

【図７】

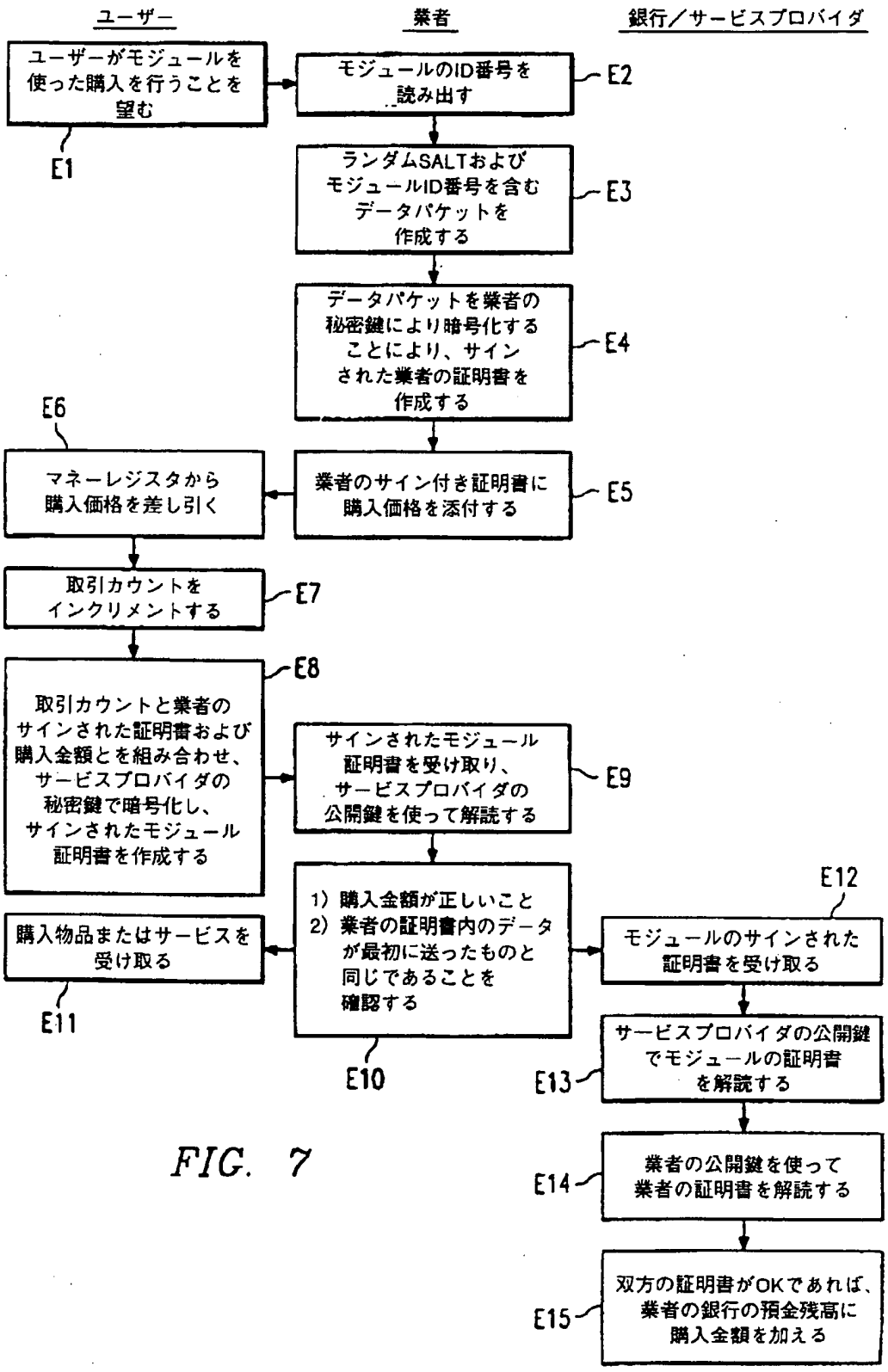

ユーザー
業者
銀行／サービスプロバイダ
ユーザーがモジュールを使った購入を行うことを望む
E1
モジュールのID番号を読み出す
E2
ランダムSALTおよびモジュールID番号を含むデータパケットを作成する
E3
データパケットを業者の秘密鍵により暗号化することにより、サインされた業者の証明書を作成する
E4
業者のサイン付き証明書に購入価格を添付する
E5
E6
マネーレジスタから購入価格を差し引く
取引カウントをインクリメントする
E7
取引カウントと業者のサインされた証明書および購入金額とを組み合わせ、サービスプロバイダの秘密鍵で暗号化し、サインされたモジュール証明書を作成する
E8
サインされたモジュール証明書を受け取り、サービスプロバイダの公開鍵を使って解読する
E9
1）購入金額が正しいこと
2）業者の証明書内のデータが最初に送ったものと同じであることを確認する
E10
購入物品またはサービスを受け取る
E11
E12
モジュールのサインされた証明書を受け取る
サービスプロバイダの公開鍵でモジュールの証明書を解読する
E13
業者の公開鍵を使って業者の証明書を解読する
E14
双方の証明書がOKであれば、業者の銀行の預金残高に購入金額を加える
E15

*FIG. 7*

【図8】

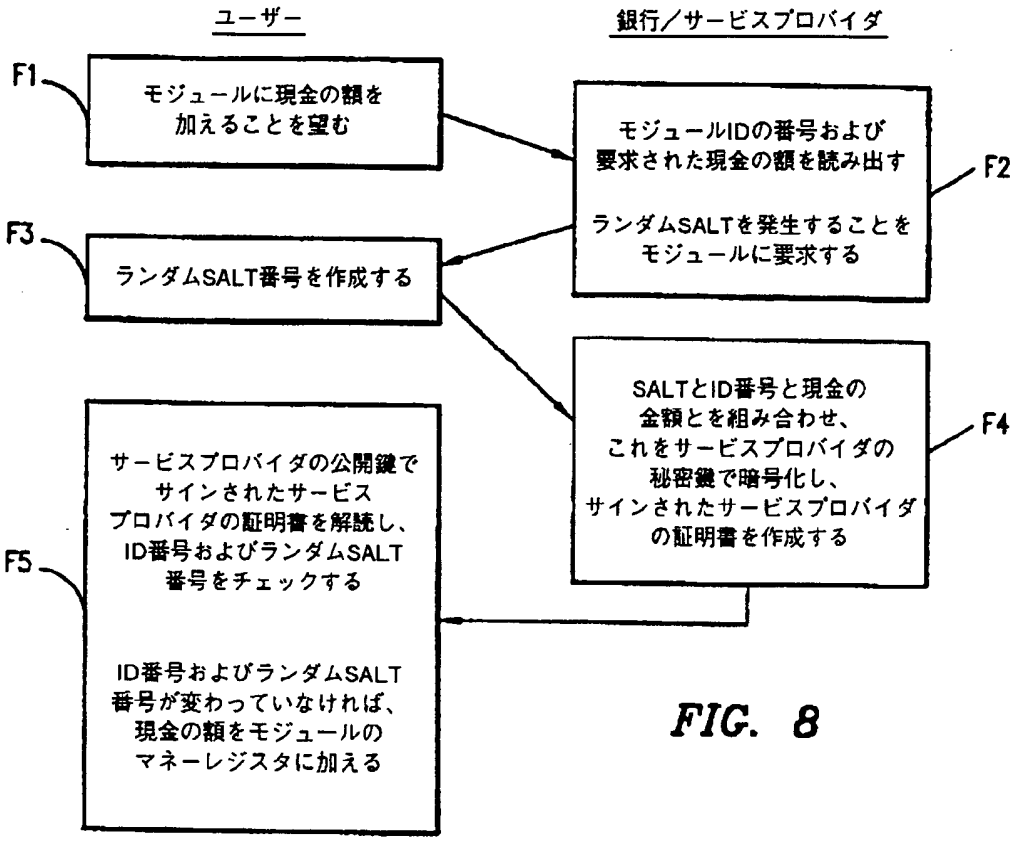

ユーザー
銀行／サービスプロバイダ
F1
モジュールに現金の額を加えることを望む
モジュールIDの番号および要求された現金の額を読み出す
ランダムSALTを発生することをモジュールに要求する
F2
F3
ランダムSALT番号を作成する
SALTとID番号と現金の金額とを組み合わせ、これをサービスプロバイダの秘密鍵で暗号化し、サインされたサービスプロバイダの証明書を作成する
F4
F5
サービスプロバイダの公開鍵でサインされたサービスプロバイダの証明書を解読し、ID番号およびランダムSALT番号をチェックする
ID番号およびランダムSALT番号が変わっていなければ、現金の額をモジュールのマネーレジスタに加える

FIG. 8

【図９】

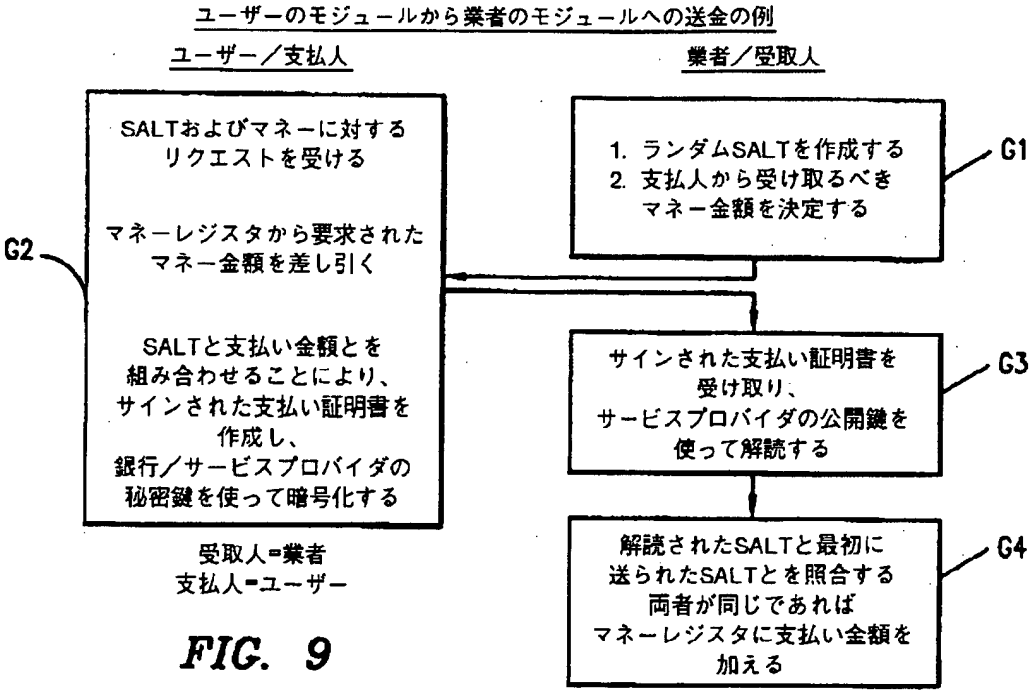
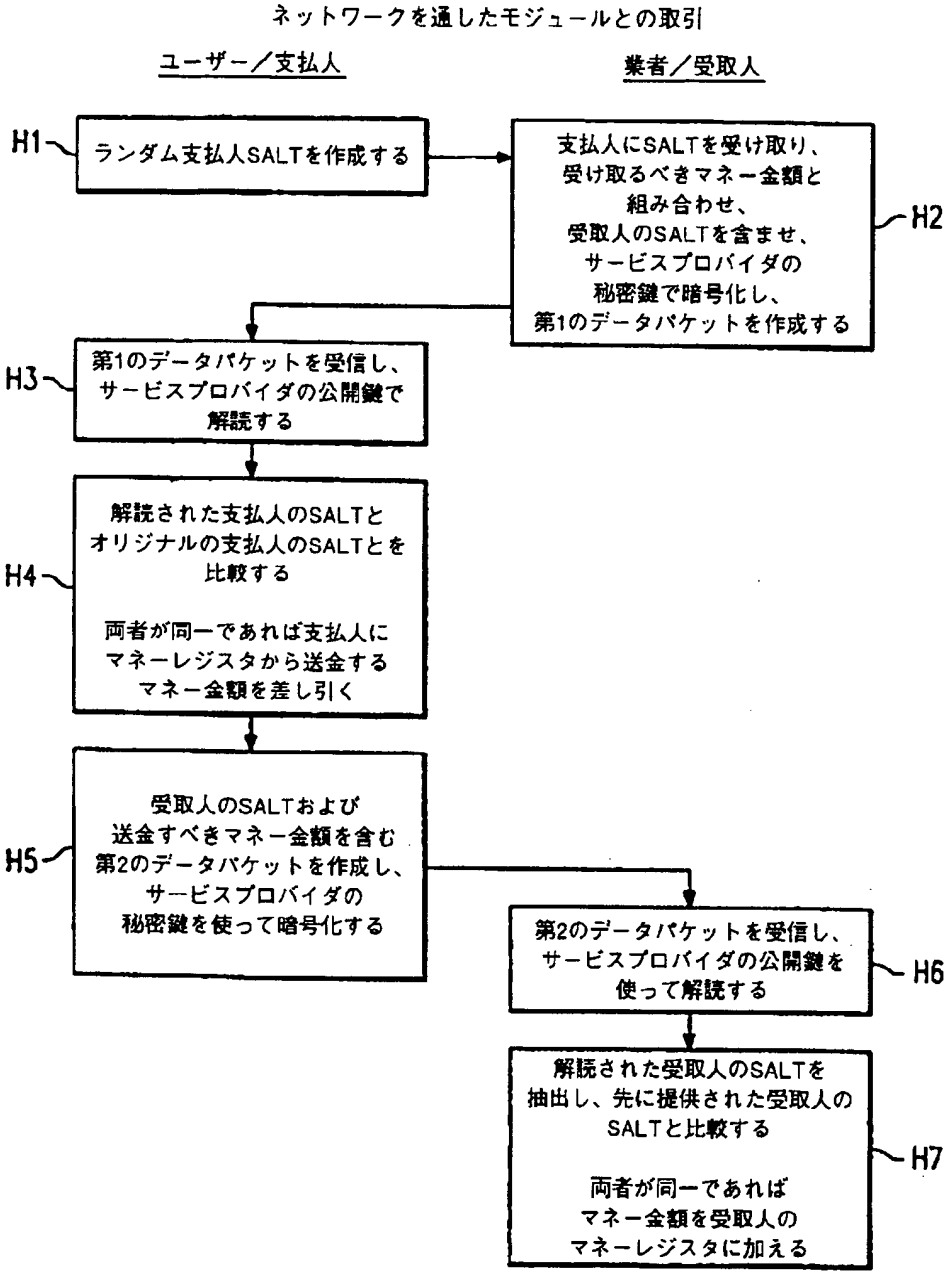

ユーザーのモジュールから業者のモジュールへの送金の例
ユーザー／支払人
業者／受取人
SALTおよびマネーに対する
リクエストを受ける
マネーレジスタから要求された
マネー金額を差し引く
SALTと支払い金額とを
組み合わせることにより、
サインされた支払い証明書を
作成し、
銀行／サービスプロバイダの
秘密鍵を使って暗号化する
受取人=業者
支払人=ユーザー
1. ランダムSALTを作成する
2. 支払人から受け取るべき
マネー金額を決定する
G1
G2
サインされた支払い証明書を
受け取り、
サービスプロバイダの公開鍵を
使って解読する
G3
解読されたSALTと最初に
送られたSALTとを照合する
両者が同じであれば
マネーレジスタに支払い金額を
加える
G4

*FIG. 9*

【図１０】

*FIG. 10*


ネットワークを通したモジュールとの取引
ユーザー／支払人
業者／受取人
H1
ランダム支払人SALTを作成する
支払人にSALTを受け取り、
受け取るべきマネー金額と
組み合わせ、
受取人のSALTを含ませ、
サービスプロバイダの
秘密鍵で暗号化し、
第1のデータパケットを作成する
H2
H3
第1のデータパケットを受信し、
サービスプロバイダの公開鍵で
解読する
H4
解読された支払人のSALTと
オリジナルの支払人のSALTとを
比較する
両者が同一であれば支払人に
マネーレジスタから送金する
マネー金額を差し引く
H5
受取人のSALTおよび
送金すべきマネー金額を含む
第2のデータパケットを作成し、
サービスプロバイダの
秘密鍵を使って暗号化する
第2のデータパケットを受信し、
サービスプロバイダの公開鍵を
使って解読する
H6
解読された受取人のSALTを
抽出し、先に提供された受取人の
SALTと比較する
両者が同一であれば
マネー金額を受取人の
マネーレジスタに加える
H7

【図１１】

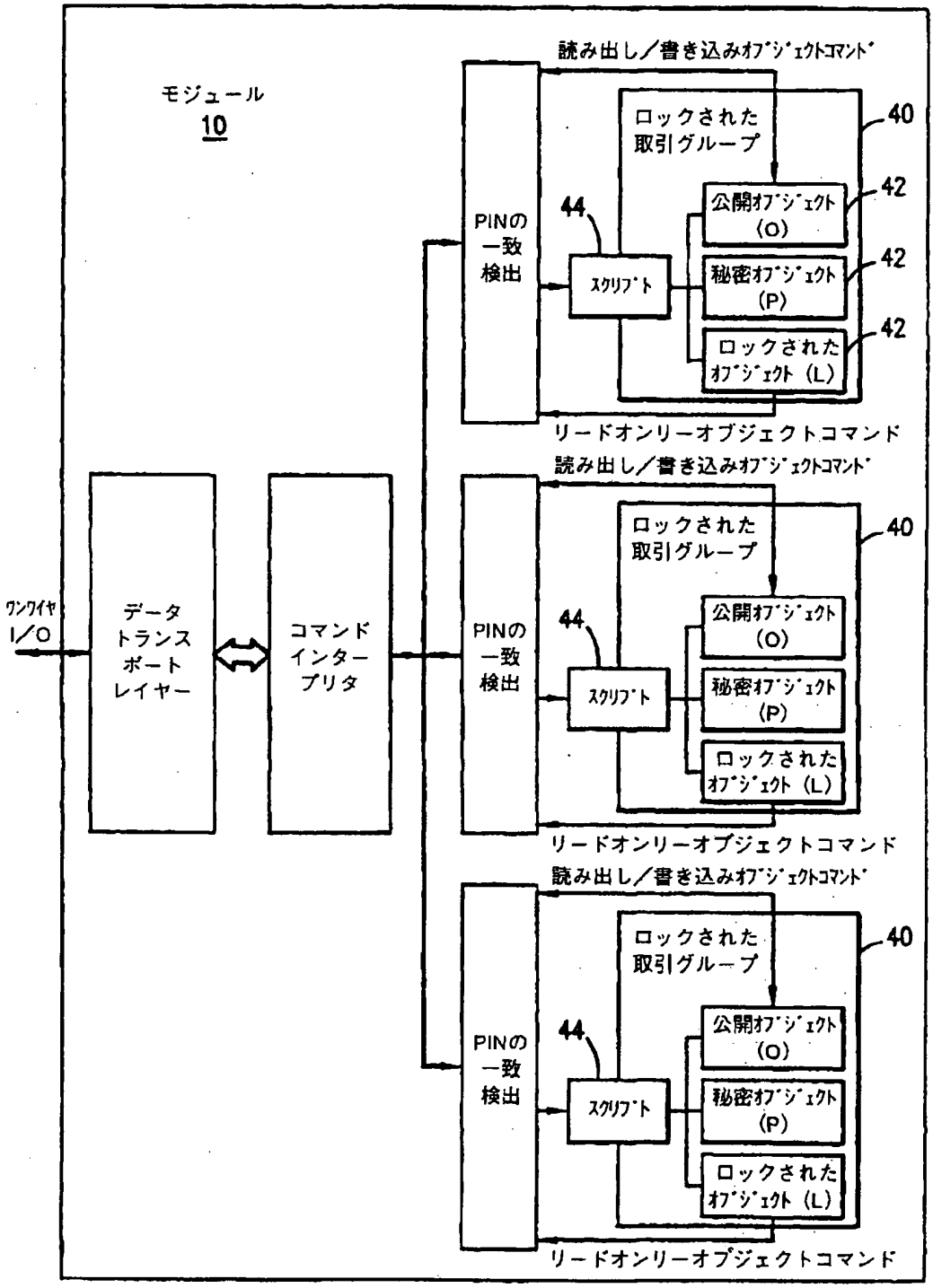

モジュール
10
ワンワイヤ
I/O
データ
トランス
ポート
レイヤー
コマンド
インター
プリタ
PINの
一致
検出
読み出し／書き込みオブジェクトコマンド
ロック された
取引グループ
40
44
スクリプト
公開オブジェクト
(O)
秘密オブジェクト
(P)
ロックされた
オブジェクト (L)
42
リードオンリーオブジェクトコマンド

FIG. 11

【図１２】

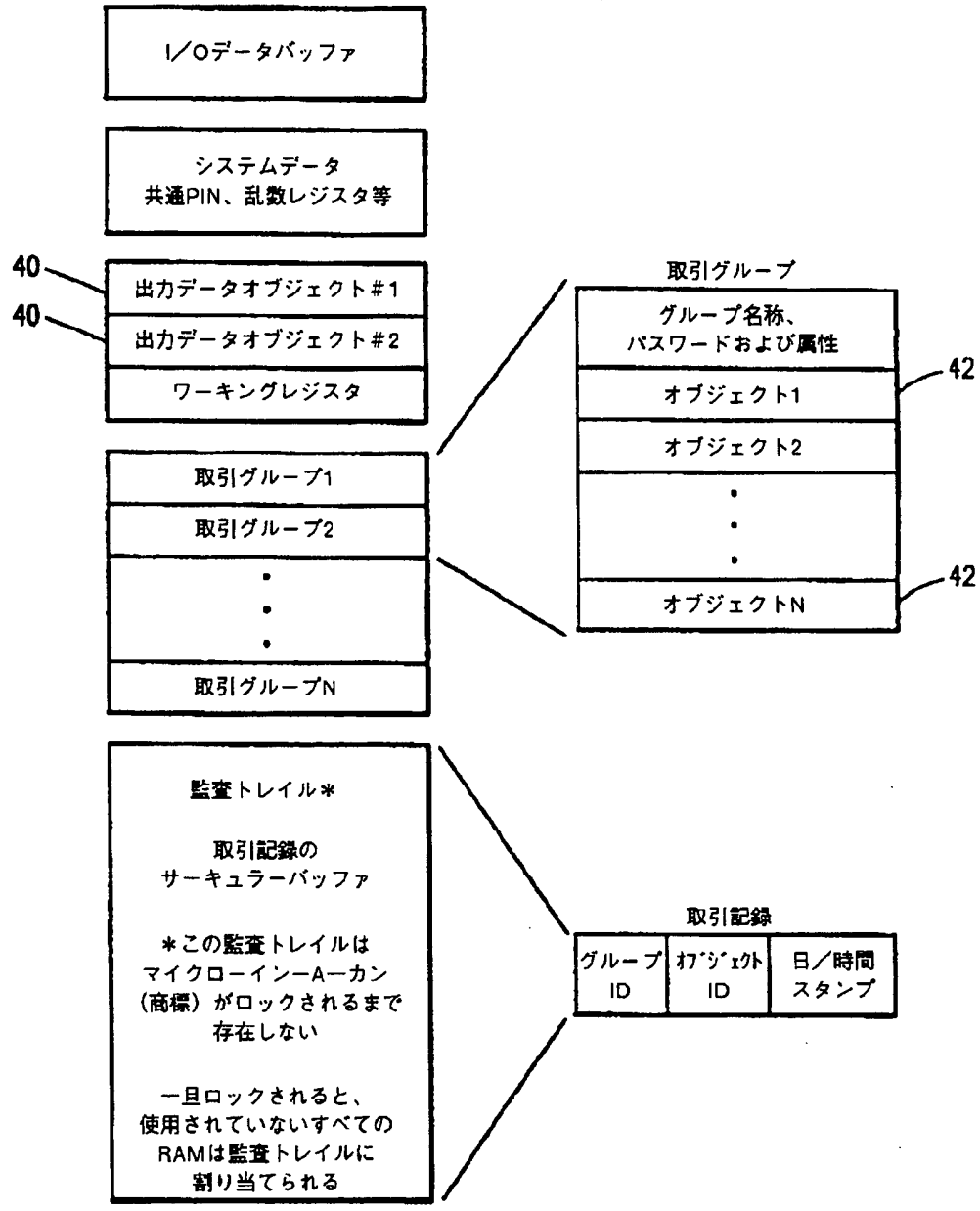

I/Oデータバッファ
システムデータ
共通PIN、乱数レジスタ等
40
出力データオブジェクト#1
40
出力データオブジェクト#2
ワーキングレジスタ
取引グループ1
取引グループ2
取引グループN
取引グループ
グループ名称、
パスワードおよび属性
オブジェクト1
オブジェクト2
オブジェクトN
42
42
監査トレイル*
取引記録の
サーキュラーバッファ
*この監査トレイルは
マイクローインーA一カン
（商標）がロックされるまで
存在しない
一旦ロックされると、
使用されていないすべての
RAMは監査トレイルに
割り当てられる
取引記録
グループ
ID
ｵﾌﾞｼﾞｪｸﾄ
ID
日/時間
スタンプ

FIG. 12

【国際調査報告】

# INTERNATIONAL SEARCH REPORT

Interna al Application No
PCT/US 96/15471

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G07F7/10 G07F19/00 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G07F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category * | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| Y | EP 0 458 306 A (TOSHIBA) 27 November 1991<br>see abstract; claims; figures 1-7<br>--- | 1,4 |
| Y<br>X<br>A | EP 0 186 981 A (IBM) 9 July 1986<br>see abstract; claims; figures 1-7<br>--- | 1,4<br>21<br>8,9,13 |
| Y | EP 0 194 839 A (TOSHIBA) 17 September 1986<br>see abstract; claims; figures<br>see page 9, line 1 - page 15, line 24<br>see page 18, line 26 - page 19, line 23<br>---<br>-/-- | 1,2,<br>6-10,<br>12-14,<br>17,18 |

[X] Further documents are listed in the continuation of box C. [X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 21 March 1997 | 02. 04. 97 |

| Name and mailing address of the ISA | Authorized officer |
|---|---|
| European Patent Office, P.B. 5818 Patentlaan 2<br>NL - 2280 HV Rijswijk<br>Tel. ( + 31-70) 340-2040, Tx. 31 651 epo nl,<br>Fax ( + 31-70) 340-3016 | David, J |

Form PCT/ISA/210 (second sheet) (July 1992)

# INTERNATIONAL SEARCH REPORT

Internat ual Application No
PCT/US 96/15471

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

| Category * | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| Y | DE 44 06 602 A (DEUTSCHE BUNDESPOST TELEKOM) 7 September 1995 | 1,2, 6-10, 12-14, 17,18 |
| A | see the whole document | 4,21 |
| A | EP 0 294 248 A (ELECTRONIQUE SERGE DASSAULT) 7 December 1988 see the whole document | 1-3,6-9, 12-20 |
| A | EP 0 624 014 A (A.M. FISCHER) 9 November 1994 see abstract; claims; figures | 1,4,5,22 |
| A | EP 0 337 185 A (SPA SYSPATRONIC) 18 October 1989 | |
| A | WO 93 08545 A (JONHIG) 29 April 1993 | |
| A | EP 0 172 670 A (TECHNION RESEARCH & DEVELOPMENT) 26 February 1986 | |

# INTERNATIONAL SEARCH REPORT

...formation on patent family members

Inter... al Application No
PCT/US 96/15471

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| EP 0458306 A | 27-11-91 | JP 4033085 A | 04-02-92 |
| EP 0186981 A | 09-07-86 | GB 2168514 A | 18-06-86 |
| | | DE 3585439 A | 02-04-92 |
| | | JP 61139878 A | 27-06-86 |
| | | US 4731842 A | 15-03-88 |
| EP 0194839 A | 17-09-86 | JP 6091526 B | 14-11-94 |
| | | JP 62000140 A | 06-01-87 |
| | | US 4862501 A | 29-08-89 |
| DE 4406602 A | 07-09-95 | NONE | |
| EP 0294248 A | 07-12-88 | FR 2615638 A | 25-11-88 |
| | | DE 3887207 D | 03-03-94 |
| | | DE 3887207 T | 26-05-94 |
| | | ES 2048211 T | 16-03-94 |
| EP 0624014 A | 09-11-94 | US 5422953 A | 06-06-95 |
| | | AU 666424 B | 08-02-96 |
| | | AU 5778194 A | 17-11-94 |
| | | CA 2120665 A | 06-11-94 |
| | | JP 7254897 A | 03-10-95 |
| EP 0337185 A | 18-10-89 | AT 123347 T | 15-06-95 |
| | | DE 58909263 D | 06-07-95 |
| | | ES 2072870 T | 01-08-95 |
| | | US 4985921 A | 15-01-91 |
| WO 9308545 A | 29-04-93 | AT 145744 T | 15-12-96 |
| | | AU 663739 B | 19-10-95 |
| | | AU 2888692 A | 21-05-93 |
| | | BR 9205416 A | 17-05-94 |
| | | CA 2098481 A | 17-04-93 |
| | | DE 69215501 D | 09-01-97 |
| | | EP 0567610 A | 03-11-93 |
| | | JP 6503913 T | 28-04-94 |
| | | PL 299825 A | 18-04-94 |
| | | US 5440634 A | 08-08-95 |

# INTERNATIONAL SEARCH REPORT

Information on patent family members

Inter nal Application No
PCT/US 96/15471

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| EP 0172670 A | 26-02-86 | JP 61094177 A | 13-05-86 |

フロントページの続き

| (51) Int.Cl.[6] | | 識別記号 |
|---|---|---|
| G09C | 1/00 | 620 |
| | | 660 |

| FI | | |
|---|---|---|
| G09C | 1/00 | 660G |
| | | 660F |
| G06F | 15/30 | M |
| | | L |
| | 15/21 | 340Z |
| | | A |

(81)指定国 EP(AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OA(BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), AP(KE, LS, MW, SD, SZ, UG), UA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN

(72)発明者 フォックス，クリストファー，ダブリュ．
アメリカ合衆国75287 テキサス州 ダラス，ティムバーグレン 3847 ナンバー4222